

DESARROLLO DE FUNCIONALIDADES WEB NO-CODE ORIENTADAS A LA
ADMINISTRACIÓN DE VACANTES PARA UNA AGENCIA DE EMPLEO

EDGAR FABIÁN LINDARTE MORA

UNIVERSIDAD DE PAMPLONA
FACULTAD DE INGENIERÍAS Y ARQUITECTURA
DEPARTAMENTO DEEST (ELÉCTRICA, ELECTRÓNICA, SISTEMAS Y
TELECOMUNICACIONES)
INGENIERÍA DE SISTEMAS
PAMPLONA, NORTE DE SANTANDER
2022

DESARROLLO DE FUNCIONALIDADES WEB NO-CODE ORIENTADAS A LA
ADMINISTRACIÓN DE VACANTES PARA UNA AGENCIA DE EMPLEO

EDGAR FABIÁN LINDARTE MORA

TRABAJO DE GRADO

Ailín Orjuela Duarte
PhD Educación
MSC Ciencias Computacionales

UNIVERSIDAD DE PAMPLONA
FACULTAD DE INGENIERÍAS Y ARQUITECTURA
DEPARTAMENTO DEEST (ELÉCTRICA, ELECTRÓNICA, SISTEMAS Y
TELECOMUNICACIONES)
INGENIERÍA DE SISTEMAS
PAMPLONA
2022

Dedicatoria

El trabajo de grado a continuación, realizado con su mucho empeño y dedicación han hecho de mí una persona completa, ética y profesional, por esto, dedico a personas como mi padre con su motivación y consejos, a mi tío, que fue como mi segundo padre y, por tanto, un apoyo fundamental, a mi madre que desde el cielo está orgullosa de mí, y mi novia que su apoyo emocional fue pieza fundamental. De forma a mi familia por cada consejo y su apoyo que me dieron para hoy día ser quien soy. De igual manera un gran agradecimiento a mis docentes quienes me apoyaron durante todo el proceso académico. Mi directora de grado Ailín Orjuela Duarte, por su gran aporte en conocimiento y que sin su colaboración no sería posible haber realizado con éxito el presente trabajo de grado. Finalmente, mis compañeros y grandes amigos en la universidad, compañeros de tantas experiencias que nos formaron de igual manera.

CONTENIDO

	Pág.
1. INTRODUCCIÓN	13
1.1 LA EMPRESA WORKI	14
1.1.1 Misión	14
1.1.2 Visión	14
1.1.3 Procesos Misionales	14
1.2 PLANTEAMIENTO DEL PROBLEMA	14
1.3 DEFINICIÓN DEL PROBLEMA	14
1.4 JUSTIFICACIÓN	15
1.5 OBJETIVOS	16
1.5.1 Objetivo General	16
1.5.2 Objetivos Específicos.....	16
1.5.3 Formulación Del Problema	16
1.5.4 Acotaciones	16
1.6 MATERIALES Y MÉTODOS	17
1.6.1 Materiales.....	17
1.6.2 Metodología	17
1.6.3 Tipo De Estudio	17
2 MARCO TEÓRICO	18
2.1 HERRAMIENTAS NO-CODE	18
2.2 BUBBLE	18
2.3 CARACTERISTICAS PARA EL DESARROLLO	19
2.3.1 Desarrollo.....	19
2.3.2 Diseño	20
2.3.3 Host	21
2.3.4 Crecimiento.....	22
2.3.5 Colaboración	23
2.4 KANBAN.....	23
2.4.1 Tablero Kanban.....	24
2.4.2 Estimación de las tareas	24
2.5 CLICKUP.....	25
2.6 ANTECEDENTES.....	26

2.6.1	Internacionales	26
2.6.2	Nacionales	27
3	DESARROLLO DEL PROYECTO	28
3.1	REQUERIMIENTOS.....	28
3.2	DIAGRAMA DE CASOS DE USO	29
3.2.1	Caso de Uso enviar candidatos.	32
3.2.2	Caso de uso recomendar sugerencias.....	33
3.2.3	Caso de uso sugerir candidatos.	34
3.2.4	Caso de uso cambiar estado	35
3.2.5	Caso de uso consultar candidatos.....	36
3.2.6	Caso de uso consultar vacantes.	37
3.2.7	Caso de uso registrar vacantes	38
3.2.8	Caso de uso validar cuenta.....	39
3.3	DESCRIPCIÓN DEL DESARROLLO	40
3.3.1	Fases del desarrollo	40
3.3.2	Fase preliminar.....	41
3.3.3	Registrar Vacante	45
3.3.4	Consultar una vacante	48
3.3.5	Consultar candidatos	49
3.3.6	Cambiar estado.....	49
3.3.7	Sugerir candidatos.....	50
3.3.8	Recomendar sugerencias.....	50
3.3.9	Enviar vacantes	51
3.3.10	Validar cuenta	51
3.4	VALIDACIÓN DEL DESARROLLO	52
3.4.1	Validación en un prototipo.....	52
3.4.2	Validación para validar cuenta.....	54
3.4.3	Validación para sugerir candidatos	55
3.4.4	Validación para recomendar sugerencias.....	57
3.4.5	Validación para enviar vacantes.....	58
3.4.6	Validación para registrar vacante.....	59
3.4.7	Validación para consultar candidatos.....	60
3.4.8	Validación para consultar vacantes	62

3.4.9	Validación para cambiar de estado	63
4	CONCLUSIONES	65
5	BIBLIOGRAFÍA	66

LISTA DE TABLAS

Tabla 1. Enviar Candidatos	32
Tabla 2. Recomendar sugerencias.....	33
Tabla 3. Sugerir candidatos.....	34
Tabla 4. Cambiar estado.....	35
Tabla 5. Consultar candidatos	36
Tabla 6. Consultar vacantes	37
Tabla 7. Registrar vacantes	38
Tabla 8. Validar cuenta	39
Tabla 9. Calificación de diseño validar cuenta	54
Tabla 10. Calificación de diseño sugerir candidatos.....	56
Tabla 11. Calificación de diseño recomendar sugerencias	57
Tabla 12. Calificación de diseño enviar vacantes	58
Tabla 13. Calificación de diseño registrar vacantes.....	59
Tabla 14. Calificación de diseño consultar candidatos	61
Tabla 15. Calificación de diseño consultar vacantes	62
Tabla 16. Calificación de diseño cambiar de estado	64

LISTA DE GRÁFICAS

	Pág.
Grafico 1. Fases de desarrollo	40

LISTA DE FIGURAS

	Pág.
Figura 1. Herramientas no code	18
Figura 2. Workflow	19
Figura 4. Tablero kanban	24
Figura 5. Tablero ClickUp	25
Figura 6. Diagrama de casos de usos	30
Figura 7. Diagrama de Clases	31
Figura 8. Diagrama de Secuencia enviar candidatos	32
Figura 9. Diagrama de secuencia recomendar sugerencias.....	33
Figura 10. Diagrama de secuencia sugerir candidatos	34
Figura 11. Diagrama de secuencia cambiar estado	35
Figura 12. Diagrama de secuencia consultar candidatos	36
Figura 13. Diagrama de secuencia consultar vacantes.....	37
Figura 14. Diagrama de secuencia registrar vacantes	38
Figura 15. Diagrama de secuencia validar cuenta.....	39
Figura 16. Editor bubble	41
Figura 17. Sección de desing	42
Figura 18. Sección workflow	43
Figura 19. Sección data	43
Figura 20. Sección styles.....	44
Figura 21. Sección de plugins	44
Figura 22. Sección de settings.....	45
Figura 23. Sección de logs	45

Figura 24. Creador de tablas	46
Figura 25. Creador de campos	46
Figura 26. Botón para Workflow	47
Figura 27. Acción para creación de tabla.....	47
Figura 28. Configuración grupo de repetición	48
Figura 29. Acción para modificar una tabla.....	50
Figura 30. Workflow para validar un tipo de usuario	52
Figura 31. Prueba de rendimiento validar cuenta.....	55
Figura 32. Prueba de rendimiento sugerir candidatos	56
Figura 33. Prueba de rendimiento recomendar sugerencias.....	57
Figura 34. Prueba de rendimiento enviar vacantes	59
Figura 35. Prueba de rendimiento registrar vacantes	60
Figura 36. Prueba de rendimiento consultar candidatos	61
Figura 37. Prueba de rendimiento consultar vacantes.....	63
Figura 38. Prueba de rendimiento cambiar de estado	64

RESUMEN

Un empleo según el perfil del trabajador, así mismo, para las empresas ayudar a conseguir el perfil correcto para las distintas vacantes. Teniendo en cuenta esto, nace la necesidad de gestionar vacantes, haciendo seguimiento desde su registro hasta la clasificación de las entrevistas, aplicando algoritmos que realicen el emparejamiento correcto entre candidato y vacante, para esto se realiza la construcción de funcionalidades de la plataforma web No-code para la administración de vacantes. De igual forma tanto para candidatos y empresas, se requieren funcionalidades para el registro y administración que ayuden a conectar con las vacantes publicadas con anterioridad. En Worki, se realiza una división interna de áreas, Placement, Expert y Atracción, el enfoque principal donde se realiza el proyecto es desde Placement, donde se realiza el enlace directo entre vacante y empresa.

Debido a que Worki Jobs es un startup¹, la empresa requiere un desarrollo ágil, usando la herramienta de desarrollo No-code, Bubble.io, y la metodología de desarrollo ágil, Kanban; por consiguiente, el presente trabajo se realizó en la modalidad de práctica empresarial que comprende el desarrollo de funcionalidades web no-code orientadas a la administración de vacantes para una agencia de empleo.

PALABRAS CLAVE: No-code, Kanban, Agencia de empleo, Vacantes, Empresas

ABSTRACT

Getting the dream job is often a difficult task, Worki is a company that is responsible for managing both companies and candidates (Workis) a job according to the profile of the worker, likewise, for companies to help get the right profile for the various vacancies. Taking this into account, the need to manage vacancies arises, tracking from registration to the classification of interviews, applying algorithms that make the right match between candidate and vacancy, for this the development for the construction of the No-code web platform for the administration of vacancies is performed. Similarly, for both candidates and companies, functionalities for registration and administration are required to help connect with previously published vacancies. In Worki, there is an internal division of areas, Placement, Expert and Attraction, the main focus where the project is carried out is from Placement, where the direct link between vacancy and company is made.

Because Worki Jobs is a startup, the company requires an agile development, using the No-code development tool, Bubble.io, and the agile development methodology, Kanban; therefore, this work was carried out in the modality of business practice comprising the development of no-code web functionalities oriented to the administration of vacancies for an employment agency.

KEY WORDS: No-code, Kanban, Employment agency, Vacancies, Companies

¹ Una Startup es una organización humana con gran capacidad de cambio, que desarrolla productos o servicios, de gran innovación, altamente deseados o requeridos por el mercado, donde su diseño y comercialización están orientados completamente al cliente.

1. INTRODUCCIÓN

El presente documento se realizó con el fin de llevar un informe del trabajo de pasantía en la empresa Worki, que requería del desarrollo de funcionalidades para gestionar datos de empleo, tanto para empresas, como para candidatos. Dada las características de la necesidad de automatizar el proceso de registro de vacantes, recomendación y selección de candidatos se seleccionó una plataforma No.code para el desarrollo de la aplicación.

Las empresas con mayores oportunidades de crecimiento, aún en medio de la situación, utilizan intermediarios como las agencias de empleo para dar soporte en el proceso de elegir a los mejores candidatos, ahorrar costos internos y automatizar sus ciclos productivos en el área de selección. De hecho, sacar provecho de los servicios de una agencia de empleo en tiempos de crisis o recesión, suele ser muy común.

Es importante resaltar que toda empresa quiere y espera buenos trabajadores con habilidades y competencias laborales múltiples. Una gran oportunidad es el acceso a una herramienta de empleo y sus diferentes funcionalidades lo que permite obtener los perfiles deseados para las diferentes vacantes de las empresas y así agilizar procesos de búsqueda y selección.

.

1.1 LA EMPRESA WORKI

1.1.1 Misión

Somos una empresa dedicada a la consultoría de desempleados con el fin de conseguir su trabajo ideal, acompañándolos desde la etapa de entrenamiento y selección hasta su posterior contratación.

1.1.2 Visión

Para el año 2025 seremos la empresa líder en consultoría de empleos por acabar con el desempleo, apoyando a las empresas a encontrar el candidato ideal, pues bien, se sabe que todos los empleadores quieren el mejor talento para su equipo.

1.1.3 Procesos Misionales

Entre estos tenemos convocatoria, selección o reclutamiento, entrevista y contratación, seguido de la motivación y capacitación.

1.2 PLANTEAMIENTO DEL PROBLEMA

Se planteó el problema inicial lo cual fue clave, y así mismo permitió dar inicio al desarrollo de la práctica, de igual forma, se expuso la justificación que son las razones por las cuales se llevó a cabo el presente proyecto.

1.3 DEFINICIÓN DEL PROBLEMA

Según una encuesta reciente realizada por la firma Adecco Colombia, reveló que para gran parte de las personas que adelantan una búsqueda activa de trabajo es muy difícil encontrar un empleo. De hecho, el 84% de los consultados considera que dar con un puesto de trabajo acorde con sus expectativas es algo particularmente complicado. Lo que permite tomar dichas estadísticas como referencia para argumentar un motivo, que facilite la búsqueda de empleo lo cual es uno de los objetivos de la compañía donde se presentó el desarrollo del proyecto.

Dentro de Worki, lugar de realización de la práctica, existen tres principales áreas, Placement, Experts y Attraction, estas tres áreas tienen distintos enfoques, pero dos objetivos principales, que el candidato (Worki) consiga su empleo soñado y que la empresa encuentre el perfil principal para la empresa. Desde Placement o colocación, allí se administran las vacantes y empresas, este proceso se hace de forma manual, es decir, el KAM (El Key Account

Manager)² realizan la administración de las vacantes y lo que requiere la empresa, estos datos de empresas, vacantes y Workis actualmente son manejados en hojas de cálculo y recolectados de manera Manual por medio de WhatsApp o redes empresariales como LinkedIn, inicialmente y con la poca demanda de vacantes y candidatos, la recolección y manejo de esta información era óptimo para el KAM, pero con el paso del tiempo y el crecimiento de la información hizo más tediosa esta recolección, generando un retraso mayor.

Sumado a lo anterior otro factor bastante imprescindible fue la carencia de tiempo existente para llevar a cabo el desarrollo del proyecto. Las empresas emergentes o startups, llevar a cabo un desarrollo lento ocasionaba una gran problemática para enfrentar un crecimiento exponencial en las inversiones extranjeras que llevan a cabo varios tipos de series de las mismas en las clasificaciones existentes, así mismo, tampoco se contaba con el recurso económico necesario para realizar contrataciones de personal especializado en tecnología y conformar un equipo completo de tecnología.

1.4 JUSTIFICACIÓN

Debido a que esta recolección y administración de estos datos eran llevadas de forma manual, se buscó sistematizar y automatizar este proceso, regulando funcionalidades como el registro de las empresas y vacantes, así mismo administrar la distinta información que se tiene de los Workis desde otras áreas para tener un mejor manejo en la selección, garantizando así, un proceso mucho más eficiente y ágil.

De igual manera, teniendo en cuenta que Worki Jobs es una empresa en crecimiento tipo Startup, se requiere que su desarrollo sea de forma ágil, las inversiones realizadas por los fondos extranjeros obligan a que el desarrollo del proyecto se lleve de forma acelerada, estas empresas necesitan ver resultados de acuerdo a ciertas metas pactadas en los contratos firmados entre los fundadores de la empresa y los inversionistas, por tal motivo se planteó que la ejecución de este desarrollo web se realice en software No-code, implementando la herramienta Bubble.io, esta herramienta ofrece un rápido desarrollo por medio de elementos gráficos que permiten desarrollar y crear páginas de forma más ágil, como también administrar distintas funcionalidades a nivel de administración de base de datos.

²Persona que se encarga de gestionar con los clientes que producen más ingresos, y desarrollan estrategias para mantener esa relación a largo plazo.

1.5 OBJETIVOS

1.5.1 Objetivo General

Realizar el desarrollo de las funcionalidades web para la gestión de candidatos y vacantes a través de la herramienta No-Code Bubble.io.

1.5.2 Objetivos Específicos

- Construir el marco teórico y estado del arte para el desarrollo de las funcionalidades para la gestión de vacantes y candidatos.
- Definir el desarrollo de las funcionalidades para la gestión de vacantes y candidatos No-code para la gestión de candidatos y vacantes.
- Validar el desarrollo de las funcionalidades para la gestión de vacantes y candidatos No-code para la gestión de candidatos y vacantes.

1.5.3 Formulación Del Problema

¿Cómo puede el desarrollo de funcionalidades web basadas en herramientas no-code ayudar a la administración de vacantes para una agencia de empleo (startup)?

1.5.4 Acotaciones

El desarrollo de este proyecto es de carácter práctico, ofreciendo a la empresa Worki Jobs S.A.S. una Plataforma para la gestión de vacantes y trabajadores, debido a la inversión y necesidad para la plataforma sea lanzada en poco tiempo, el desarrollo se realiza en la herramienta No-Code llamada Bubble.io, se plantea una futura transición a código debido a las limitaciones en temas de optimización de la plataforma y reducción de costos, Bubble ofrece un Sistema integrado para el desarrollo, tanto a nivel de Backend³ como de Frontend⁴. No se mostrarán fragmentos del código implementado por políticas de privacidad de la empresa.

El proyecto se desarrolló de manera remota, cuya sede principal de la empresa está ubicada en la ciudad de Bogotá, se estimó una duración aproximada del proyecto de 8 meses, comenzando en el mes de agosto y culminando en el mes de diciembre.

³ Parte del desarrollo web que se encarga de que toda la lógica de una página web funcione.

⁴ Parte del desarrollo web que se dedica a la parte frontal de un sitio web, en pocas palabras del diseño de un sitio web, desde la estructura del sitio hasta los estilos como colores, fondos, tamaños hasta llegar a las animaciones y efectos.

1.6 MATERIALES Y MÉTODOS

A continuación, se describen los materiales y metodología que fueron necesarios en la realización del desarrollo web No-code del presente proyecto.

1.6.1 Materiales

Los materiales o recursos que se implementaron a lo largo de la realización del desarrollo de funcionalidades web no-code orientadas a la administración de vacantes para una agencia de empleo⁵, se emplea la metodología Kanban, son los siguientes:

- Equipo de computo
- Software Bubble.io
- Software Slack
- Software Figma

1.6.2 Metodología

El desarrollo del presente proyecto tiene un enfoque cualitativo, ya que permite la recolección de información mediante diferentes métodos como la consulta de fuentes bibliográficas o manuales lo cual facilito la obtención detallada de la misma y una vez realizar la aplicación y desarrollo de los objetivos propuestos.

1.6.3 Tipo De Estudio

Se empleo un método de investigación descriptivo ya que se detalla los diferentes desarrollos a lo largo del proyecto, lo cual permitió obtener las conclusiones finales.

⁵ Institución que lo representa al trabajador frente a los empleadores del área de recursos humanos de la empresa que va a requerir de los servicios.

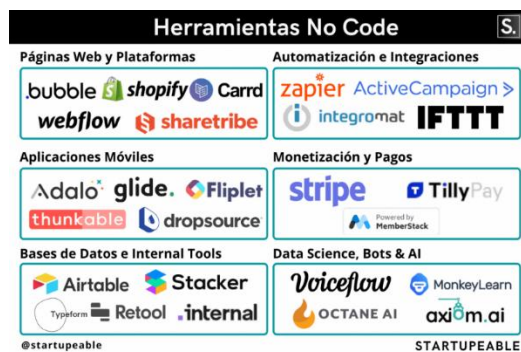
2 MARCO TEÓRICO

En esta sección se va a describir detalladamente los distintos conceptos que se aplicaron durante la realización del proyecto. Se comenzará adentrándose un poco en lo que es el mundo No-Code, esto, para poder entender un poco sobre la plataforma en que se construyeron las funcionalidades de las cuales se harán referencia durante el desarrollo del proyecto. Así mismo, se describirá un poco la metodología que se usó para hacer el levantamiento de los requerimientos y definición de tareas, así como el software implementado para administrarlo. Finalmente, también se trató algunos conceptos para describir el ambiente laboral en el que se basa el desarrollo, así como el negocio que se tiene como objetivo.

2.1 HERRAMIENTAS NO-CODE

Según el sitio Web (Startupeable): define que el desarrollo de software a lo largo de los años se ha venido realizando desde un nivel de código que se entiende como nivel máquina, en este nivel se implementan lenguajes de bajo nivel, con el pasar de los años cada vez se han implementado lenguajes de alto nivel, estos lenguajes hacen que cada vez se interactúe menos a nivel de máquina y más a nivel de usuario. No obstante, en la última década se ha experimentado el inicio de lo que podría marcar el rompimiento de tener que escribir código para desarrollar una aplicación. Aquí es donde nacen las herramientas No-Code, que ya posee elementos que han sido programados previamente, y que está desarrollando una aplicación va a visualizar el elemento que necesita para su propósito, las herramientas No-Code tienen un limitante aún muy grande para este presente, y es la optimización, muchos de estos elementos tienen código que puede ser redundante si se mezcla entre los mismos, por esto, puede causar alteraciones en el rendimiento de la máquina.

Figura 1. Herramientas no code



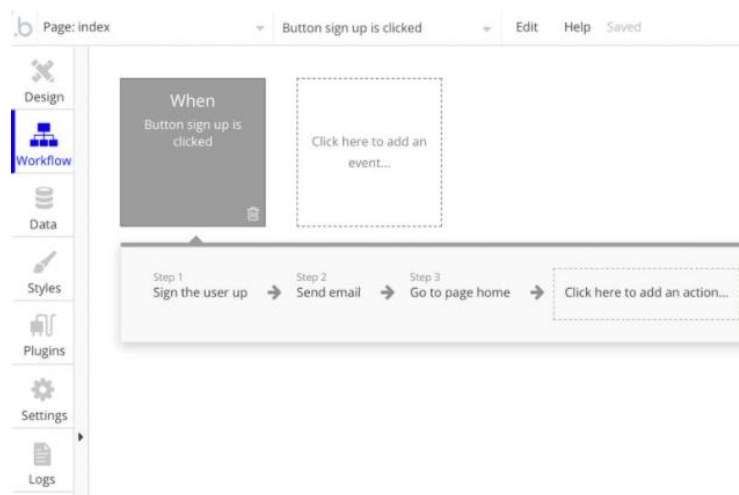
Fuente: (Startupeable)

2.2 BUBBLE

(Perez, 2021) Define a Bubble como el software en pleno apogeo que levanta aplicaciones web sin código. Es de conocimiento que construir tecnología es lento y caro, pero Bubble

permite crear apps sin tener conocimientos previos de desarrollo. Hace posible crear aplicativos interactivos y multiusuario para navegadores web de escritorio y móviles, incluyendo todas las características necesarias para construir una red social, inclusive.

Figura 2. Workflow



Fuente: (Perez, 2021)

2.3 CARACTERISTICAS PARA EL DESARROLLO

En esta sección se reflejan las principales características del desarrollo y sus funcionalidades teniendo como referencia el Manual de funciones de Buuble, las cuales se detallan a continuación:

2.3.1 Desarrollo

Totalmente funcional y personalizable, permite mapear la experiencia del usuario para determinar lo que ocurre mientras los usuarios navegan por tu aplicación, clic a clic.

Lógica potente, Construye una lógica if-then⁶ y gestiona una base de datos con nuestra herramienta de flujo de trabajo de arrastrar y soltar totalmente personalizable.

Gestión de usuarios, Configura cuentas de usuario y permite el inicio de sesión con contraseñas o con cualquier proveedor OAuth 2.0, como Facebook, LinkedIn o Google.

⁶ Es un concepto para llevar a cabo acciones basadas en distintas condiciones.

Almacenamiento de datos, Diseña propias estructuras de datos en el almacén de datos alojado, escalable y sin código, se conecta a una propia base de datos SQL o API RESTful⁷.

Datos privados, crea reglas de privacidad para gestionar los datos que los usuarios pueden ver.

Exportación y gestión de datos, importa y exporta todos los datos de la aplicación mediante archivos CSV o Excel.

Interactividad en tiempo real, Bubble envía todas las acciones de datos a otros usuarios en tiempo real, por lo que es fácil crear chats, noticias y otros casos de uso interactivo.

Búsqueda sofisticada, Crea búsquedas y consultas avanzadas y muestra los resultados en el formato que desee. Personaliza aún más las consultas con la potente integración nativa de Algolia

Bubble con Algolia, acciones automatizadas, ahorra tiempo y trabajo manual programando acciones futuras y ejecutando flujos de trabajo recurrentes para automatizar los procesos de backend.

Infinitamente extensible, amplía completamente la aplicación Bubble con plugins de Javascript y nuestro conector API⁸ sin código. Encuentra una solución a cualquier problema empresarial en el mercado de plugins, o permite crear los que el usuario desee.

Vistas de base de datos personalizadas, crea vistas personalizadas de la base de datos con restricciones y filtros para obtener información sobre los datos de su aplicación o se podrá utilizar como fuente de datos para los flujos de trabajo del backend para realizar operaciones de datos aún más potentes.

Conjuntos de opciones (Option sets), crea y almacena fácilmente listas globales de opciones para la aplicación fuera de la base de datos. Los conjuntos de opciones son una excelente forma de definir desplegables o variables con atributos que pueden utilizarse en toda la aplicación.

2.3.2 Diseño

Editor de arrastrar y soltar, crea un diseño perfecto sin conocimientos de HTML o CSS.
Diseño responsivo, las aplicaciones Bubble se ven muy bien en los navegadores móviles, desde el primer momento ya que Bubble posee un diseño h.

⁷ Es una interfaz que dos sistemas de computación utilizan para intercambiar información de manera segura a través de Internet.

⁸ Define las reglas que se deben seguir para comunicarse con otros sistemas de software.

Dominios personalizados, crea una marca blanca para tu aplicación con un nombre de dominio personalizado, optimizado con Cloudflare⁹ para un rendimiento más rápido.

Estilo visual, inserta fácilmente imágenes, iconos, botones, mapas, vídeos, calendarios y mucho más.

Contenido dinámico, el texto y las imágenes pueden mostrar contenido generado por el usuario o datos obtenidos de APIs u otros servicios.

Control condicional, cambia la fuente, el color o la visibilidad de cualquier elemento de la página en función de la información de la aplicación.

Plugins¹⁰, muestra información de otras aplicaciones con nuestra colección de más de 800 plugins, incluyendo Stripe, Google, Facebook y otros.

Internacionalización fácil, traduce el texto de su aplicación y muestre automáticamente el idioma, la moneda y los formatos correctos sin trabajo adicional.

Estilos para toda la aplicación, define estilos para toda la aplicación para mantener un diseño coherente en toda la aplicación.

Integración con Figma¹¹, importa los marcos¹² en Bubble y convierte los diseños en una aplicación funcional en muy poco tiempo.

2.3.3 Host

Implantación y alojamiento sencillos, no requiere de administración en sus centrales, la infraestructura o las operaciones del servidor.

Escala con facilidad, asigna más recursos de servidor a medida que crece la base de usuarios para pasar de una aplicación de demostración a una empresa a escala web sin cambios técnicos.

⁹ Es una red de entrega de contenido.

¹⁰ Complementos que añaden funcionalidades extra o mejoras a los programas.

¹¹ Es una herramienta de prototipado web y editor de gráficos vectorial que, a diferencia de las otras herramientas, se aloja en la web.

¹² Herramienta creada para simplificar la vida de los desarrolladores ofrecen estructuras y configuraciones de entorno para agilizar el proceso de desarrollo.

Control de versiones, prueba los cambios de forma segura en una versión privada y, a continuación, implementa para sus usuarios con un solo clic. Revierte instantáneamente su aplicación a cualquier punto de su historia.

Copias de seguridad automáticas, Bubble realiza copias de seguridad de los datos de los usuarios y puede retroceder a cualquier punto en el tiempo.

Registro en tiempo real, ve cada acción de los usuarios en la aplicación y traza un historial de su comportamiento.

Es global, Bubble utiliza Cloudflare¹³ CDN para todas las aplicaciones, publicando el contenido globalmente para un rápido acceso local en todo el mundo. También ofrece alojamiento internacional con planes dedicados.

Siempre seguro, mantenga a sus colaboradores y usuarios seguros con certificados SSL gratuitos y automáticos para todos los sitios web de Bubble.

Autenticación de múltiples factores, habilita en los planes de producción y dedicados para proteger su cuenta de Bubble, y permita a los usuarios proteger sus cuentas dentro de su aplicación.

Análisis incorporados, toma decisiones informadas, realiza un seguimiento del número de visitantes y de las ejecuciones del flujo de trabajo y vea qué partes de su aplicación son las más utilizadas con nuestros potentes paneles.

2.3.4 Crecimiento

Integraciones de seguimiento de la actividad, conecta con servicios populares de análisis y gestión de clientes como Mixpanel, Google Analytics o Segment¹⁴.

Correo electrónico, enviar correos electrónicos personalizados y transaccionales a los usuarios y los integra con servicios de gestión de listas de correo electrónico.

Integraciones de pago, Bubble se integra con Stripe y Braintree¹⁵ y admite modelos de pago complejos, como suscripciones, pagos diferidos, transacciones de terceros, etc.

¹³ Es un sistema gratuito que actúa como un proxy (intermediario) entre los visitantes del sitio y el servidor.

¹⁴ Mixpanel es un indicador que mide el compromiso y páginas vistas. Hace que su producto mejore por las acciones de medición y no por las páginas vistas.
Google Analytics es una herramienta de analítica web de la empresa Google. Ofrece información agrupada del tráfico que llega a los sitios web.

¹⁵ Plataformas de pago digitales.

SEO¹⁶, controla la forma en que Google muestra la página en los resultados de búsqueda estableciendo metaetiquetas, personalizando el texto y los títulos de las URL y generando mapas del sitio.

Integración con Bubble, puede publicar un API con un solo clic para que los clientes y socios se conecten con el creador.

URLs de páginas personalizables, establece los propios slugs¹⁷ para una experiencia más agradable por los usuarios.

2.3.5 Colaboración

Hasta 40 colaboradores, permite el desarrollo en conjunto, con permisos y derechos de acceso personalizables.

Actualizaciones del editor en tiempo real, permite ver lo que los colaboradores están modificando en directo para trabajar en equipo o evitar sobreponer el editor.

Anotaciones, proporciona retroalimentación marcando la aplicación con notas, tareas y comentarios para que el equipo los vea.

Desarrollo de aplicaciones privadas, protege la aplicación con una contraseña para poder compartirla de forma selectiva hasta que esté listo para subirla a la web.

2.4 KANBAN

La metodología Kanban es la manera de gestionar el trabajo de forma fluida. Proviene de Japón, Kanban es un distintivo visual que se usa para denotar una acción. Frecuentemente se representa en un tablero tradicional para exponer los procesos de su flujo de trabajo.

Una tarea, ilustrada por una tarjeta Kanban, se moverá por medio de las diversas etapas de su trabajo hasta su culminación. De forma frecuente se habla de Kanban como un método de extracción, de forma que se arroja las tareas a través del flujo de trabajo, ya que permite a los usuarios trasladar de sitio libremente las tareas en un ambiente de trabajo basado en el equipo.

¹⁶ Search Engine Optimization (Optimización para motores de búsqueda). Se trata del conjunto de estrategias y técnicas de optimización que se hacen en una página web para que aparezca orgánicamente en buscadores de Internet.

¹⁷ Es una herramienta que corresponde a la parte de la URL que identifica el contenido de tu página web.

Figura 3. Tablero kanban



Fuente: (Digitalguide, Ionos)

2.4.1 Tablero Kanban

El establecimiento de las columnas en un tablero Kanban no está 100% definido, no obstante, existen tres columnas básicas, Pendiente, En progreso y Terminado. Sin embargo, el esquema usado en este proyecto es:

- **Historias de Usuario**, aquí se describe cada actividad planeada por realizar, son las tareas que llegan de fase de descubrimiento o levantamiento del requerimiento para solucionar la necesidad.
- **Pendiente**, en esta columna se describen las tareas que están a punto de ser listas para ser ejecutadas.
- **En progreso**, en este punto de la columna se agregan todas las tareas que están siendo ejecutada.
- **Pruebas**, en esta sección del tablero se ejecutan las pruebas del desarrollo realizado, es la etapa previa al lanzamiento, se podría decir que las tareas que se encuentren allí están prácticamente terminadas.
- **Cerradas**, finalmente en esta columna están todas las tareas completamente terminadas y lanzadas al un entorno en producción o en vivo.

2.4.2 Estimación de las tareas

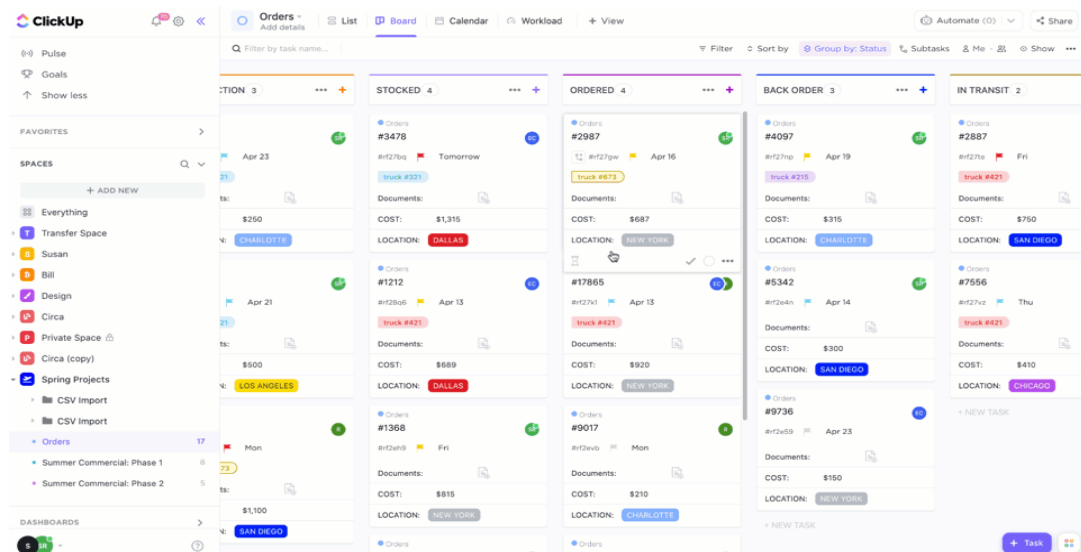
Cada tarea, o en efecto tarjeta, tiene una estimación de tiempo dada en horas, esta estimación generalmente la establece el Product Manager¹⁸ o Product Owner¹⁹, pero también se suelen establecer estos tiempos entre los anteriormente mencionados y los desarrolladores designados a la solución de las tareas. Otro punto a tener en cuenta es la prioridad de las tareas, éstas se establecen según la importancia que represente en el avance en el desarrollo.

2.5 CLICKUP

ClickUp es un software para la gestión de proyectos y colaboración orientada a gerentes miembros de equipos de desarrollo, marketing o ventas. La plataforma trae consigo tres vistas distintas para gestionar proyectos, configuraciones de tareas de forma personalizada y filtrado de proyectos por separado.

Las tres vistas diferentes de proyectos de ClickUp incluyen la vista de tablero, orientada a la metodología Kanban, y una vista ágil diseñada específicamente para la metodología SCRUM. Los administradores, Product Manager o Product Owner pueden asignar tareas a varios miembros del equipo a la vez, de igual forma personalizar en su totalidad los flujos de trabajo y prioridades desde una única página de configurable.

Figura 4. Tablero ClickUp



Fuente: (Consulting, Snoop)

¹⁸Es la figura que tiene la responsabilidad de identificar las necesidades de los consumidores y satisfacerlas a través del desarrollo y entrega de producto.

¹⁹ Profesional encargado de maximizar el valor de una solución generada en las empresas a partir de las metodologías ágiles.

2.6 ANTECEDENTES

2.6.1 Internacionales

(Woo, 2020), The Rise of No/Low Code Software Development—No Experience Needed? Principal Objetivo: rastrear cómo el coronavirus (COVID-19) había estado arrasando la ciudad de New York y brindar servicios de manera eficiente a los necesitados. Para lograr esto, crearon un portal en línea para recopilar información sobre personas con COVID-19 y otras personas en contacto con personas enfermas por el virus. La velocidad fue importante, y el Portal de participación de COVID-19 estuvo en funcionamiento en tan solo tres días, todo sin escribir ninguna una sola línea de código.

Esta hazaña, que poco después se replicó para el Centro de soporte de COVID-19 de Washington, DC, fue posible gracias a una plataforma de desarrollo de software No-Code, creada por Unqork, una empresa emergente con sede en la ciudad de Nueva York. El uso de tales plataformas sin código, y sus primos de No-code, que requieren un conocimiento mínimo de codificación, está en aumento, una tendencia que representa un paso hacia un objetivo de décadas en informática para automatizar la codificación. En este antecedente Woo, coincide en un aporte claro que fundamenta el presente trabajo de grado: el corto tiempo para llevar a cabo un desarrollo, así como se necesitó llevar a cabo la construcción de un portal web con información necesaria para el Coronavirus, de igual forma en el presente proyecto, la necesidad de negocio para levantar en corto tiempo una inversión y tener resultados más óptimos en el registro, recomendación y envío de candidatos, fue un éxito

(Hallberg, 2021) Using Low-Code Platforms to Collect Patient-Generated Health Data: A Software Developer's Perspective, Principal Objetivo: Crear una aplicación para recopilar datos de salud generados por pacientes. El hecho de que las personas recopilen sus datos de salud a través de aplicaciones de salud en sus teléfonos inteligentes se está volviendo cada vez más popular. Aun así, es difícil para los proveedores de atención médica utilizar estos datos de salud generados por los pacientes, ya que las aplicaciones de salud no pueden compartir fácilmente sus datos con los registros médicos electrónicos (EHR, por sus siglas en inglés) de los proveedores de atención médica. Simultáneamente, se está volviendo cada vez más popular el uso de plataformas low-code para el desarrollo de software. Esta tesis exploró el uso de plataformas de No-code para crear aplicaciones destinadas a recopilar datos de salud generados por pacientes y enviarlos a EHR²⁰ mediante la creación de un prototipo de aplicación web con las plataformas de No-code Mendix y Better EHR Studio(que son). Durante el desarrollo del prototipo de la aplicación web, el desarrollador llevó a cabo un diario para capturar sus impresiones de Mendix para mostrar cómo un desarrollador experimenta el desarrollo en una plataforma de No-code en comparación con la programación tradicional. El resultado muestra que no es práctico crear aplicaciones destinadas a recopilar datos de salud generados por pacientes con las dos plataformas de No-

²⁰ Es un sistema para hospitales y clínicas que almacena de manera cronológica y en formato digital todos los aspectos relativos a la salud de un paciente.

code elegidas. El análisis del diario realizado mostró que usar una plataforma de No-code es sencillo, pero también desafiante para un desarrollador de software experimentado. El aporte que da Hallberg para el proyecto, fue el mismo experimentado por la persona que llevó a cabo el desarrollo del presente proyecto, ya que los conocimientos previos eran de un nivel más alto y tradicional en el desarrollo de Software, y tuvo que adaptarse a un desarrollo más sencillo y quizás algo futurista para lógicas actuales.

2.6.2 Nacionales

(Fernandez, 2020) Las plataformas digitales, la productividad y el empleo en Colombia, Principal Objetivo: Exponer la importancia de las plataformas digitales para la productividad y el empleo en Colombia. En el ámbito local se experimenta, recientemente, debido a la pandemia, un uso de plataformas para conseguir empleo en Colombia, tal como se menciona en el reporte de julio de 2020 de Fedesarrollo, “Las plataformas favorecen la innovación, la bancarización, el uso de tecnologías, la formalización y en particular, el pago de impuestos por parte los aliados”. El aporte de Fernández a este proyecto demuestra la importancia que impacta a la sociedad las tecnologías en la búsqueda de empleo, este, es un objetivo corporativo de la empresa donde se desarrolla el presente proyecto.

(Parra Arevalo, 2020) La capacidad de customizar una aplicación con Low-code/Nocode varía de plataforma en plataforma, en algunos casos las opciones de personalización son más reducidas en algunas plataformas que en otras. Bloqueo de los vendedores: Hay proveedores que generan código que es fácil de mantener por fuera de la plataforma e incluso podría ser mantenido en una plataforma externa. Por otra parte, también existen proveedores que limitan el mantenimiento de una aplicación a una plataforma, lo que dificulta el proceso de desarrollar una aplicación. Tal como lo expone (Parra Arevalo, 2020), se encontraron limitantes a la hora de desarrollar el presente proyecto, ya que existen limitantes de capas de personalización visual, así mismo al querer usar el código base que se proyectó para poder implementarlo para futuros desarrollos de forma optimizada, es imposible extraerlo, ya que el proveedor lo impide.

3 DESARROLLO DEL PROYECTO

En esta unidad del proyecto se dará una exposición de los pasos del desarrollo web No-code para construir las funcionalidades para la administración de vacantes en una agencia de empleo, de igual forma también se expondrán los casos de uso de los flujos más importantes del desarrollo.

3.1 REQUERIMIENTOS

A continuación, se expondrán los distintos requerimientos planteados durante para la ejecución del proyecto, así mismo un análisis del porqué de cada uno.

- **Requerimiento:** El reclutador podrá ingresar los datos de la vacante y registrarlos en el sistema, los cuales podrán usarse para la recomendación de vacantes.
- **Justificación:** debido a la forma manual que era llevada a cabo la recolección de datos de los distintos reclutadores para el registro de vacantes.
- **Requerimiento:** El reclutador visualizará el listado de vacantes registradas y su cantidad de recomendaciones.
- **Justificación:** debido al alto flujo de consultas a los empleados de Worki, sobre el estado de sus vacantes, es necesario facilitar una funcionalidad que permita al reclutador consultar sus vacantes.
- **Requerimiento:** El reclutador visualizará el listado de recomendaciones enviadas para contactar candidatos o cambiarlos de estado.
- **Justificación:** semejante al requerimiento anterior, debido al alto flujo de preguntas hacía los empleados sobre los candidatos que les habían sido recomendados, se necesitaba una funcionalidad que les permitiera abrir la vacante y ver los candidatos.
- **Requerimiento:** El reclutador podrá cambiar el estado de una recomendación según su determinación.
- **Justificación:** debido a la carencia de análisis y rastreo de información sobre el estado de los candidatos luego de enviados al reclutador, se necesitaba una funcionalidad que permitiera al reclutador cambiar de estado de la vacante, para saber si este fue contactado, agendado, favorito o ha sido descartado.
- **Requerimiento:** El empleado podrá recomendar las sugerencias previamente descritas realizadas.
- **Justificación:** se tenían almacenadas las recomendaciones en hojas de cálculo y manualmente se enviaban las recomendaciones realizadas por WhatsApp, se requería una funcionalidad para enviar recomendaciones estas sugerencias recolectadas.
- **Requerimiento:** El empleado podrá enviar las recomendaciones realizadas.

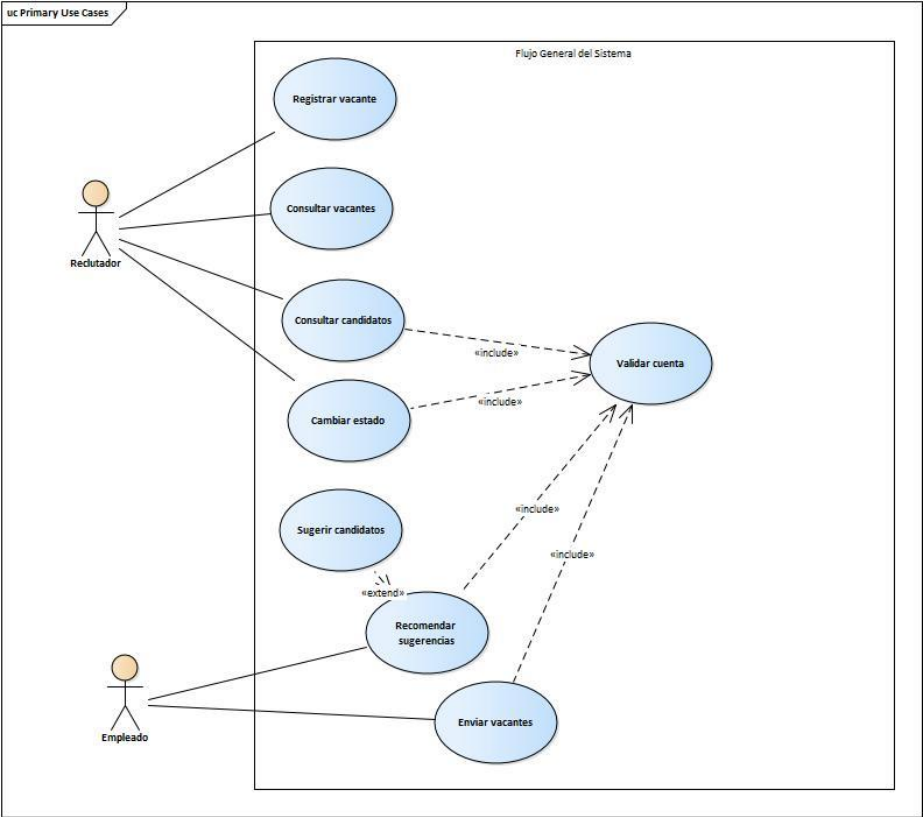
- Justificación: basado en el requerimiento anterior, se requería una forma más ágil para que las recomendaciones obtenidas, fueran enviadas de forma automatizada por medio del contacto suministrado por el reclutador.
- Requerimiento: El empleado podrá sugerir candidatos a las vacantes registradas por los reclutadores.
- Justificación: si bien las sugerencias de candidatos pueden darse desde otra área y otra funcionalidad, es posible que esta área de la empresa pueda realizar sugerencias de candidatos por medio de la funcionalidad suministrada, esta funcionalidad no existía, por tanto se requería.
- Requerimiento: El empleado y reclutador ingresan sus credenciales y podrán validar su cuenta.
- Justificación: para tener un orden y seguridad dentro de la plataforma, se requería de una plataforma para validar el ingreso, por esto, se debía realizar un ajuste de redireccionamiento a la funcionalidad de login²¹.

3.2 DIAGRAMA DE CASOS DE USO

En esta parte del proyecto se reflejan los más importantes casos de uso para la realización del desarrollo de funcionalidades web no-code orientadas a la administración de vacantes para una agencia de empleo.

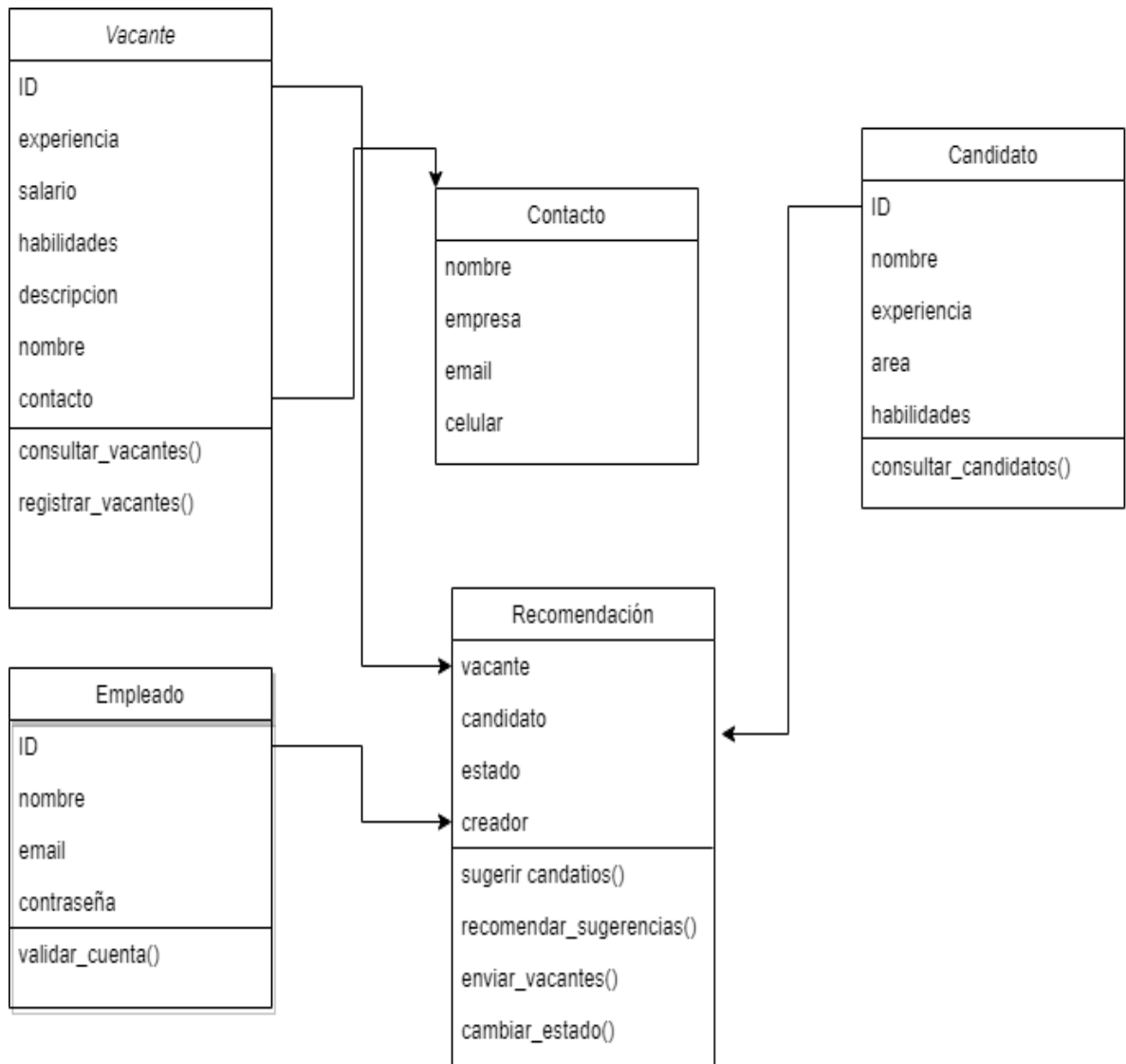
²¹ Proceso mediante el cual se controla el acceso a un sistema informático.

Figura 5. Diagrama de casos de usos



Fuente: Elaboración Propia

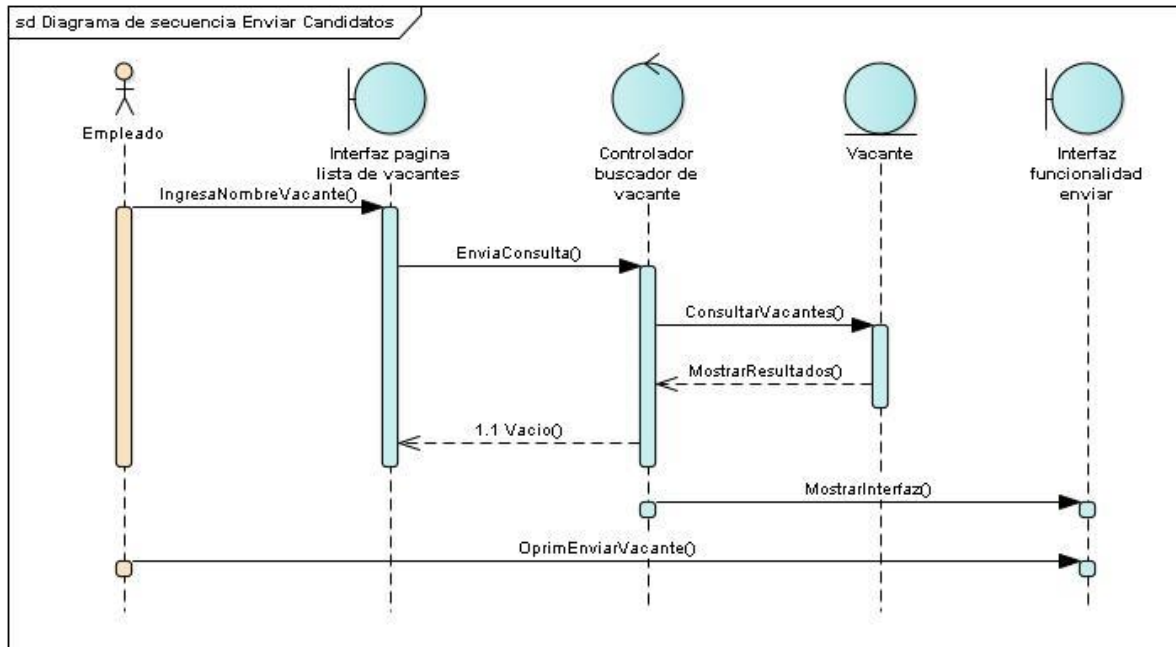
Figura 6. Diagrama de Clases



Fuente: Elaboración Propia

3.2.1 Caso de Uso enviar candidatos.

Figura 7. Diagrama de Secuencia enviar candidatos



Fuente: Elaboración Propia

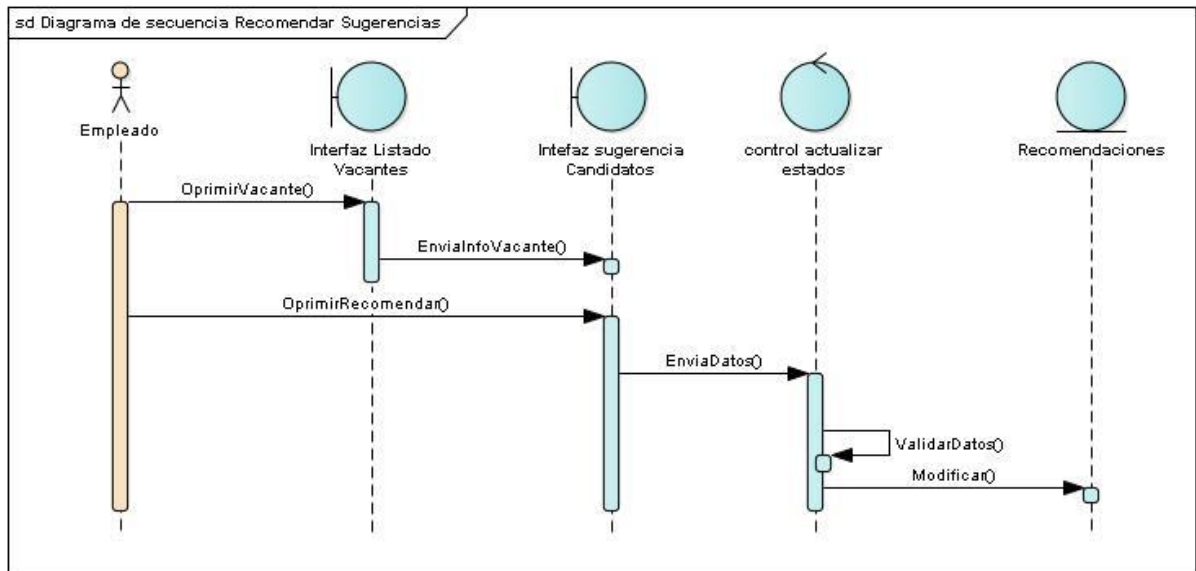
Tabla 1. Enviar Candidatos

Nombre del caso de uso	Enviar Candidatos
Actor Participante	Iniciado por el Empleado
Condición inicial	1. El usuario inicia la página principal la cual cargará el listado de vacantes
Flujo de eventos	2. Oprime el botón de Vacantes Nuevas 3. Ingresa el nombre de la vacante en el buscador 4. Se muestran los resultados de las coincidencias por nombre de la vacante 5. Oprime el botón para enviar la vacante
Condición de salida	6. La Vacante es enviada correctamente

Fuente: Elaboración propia

3.2.2 Caso de uso recomendar sugerencias.

Figura 8. Diagrama de secuencia recomendar sugerencias



Fuente: Elaboración propia

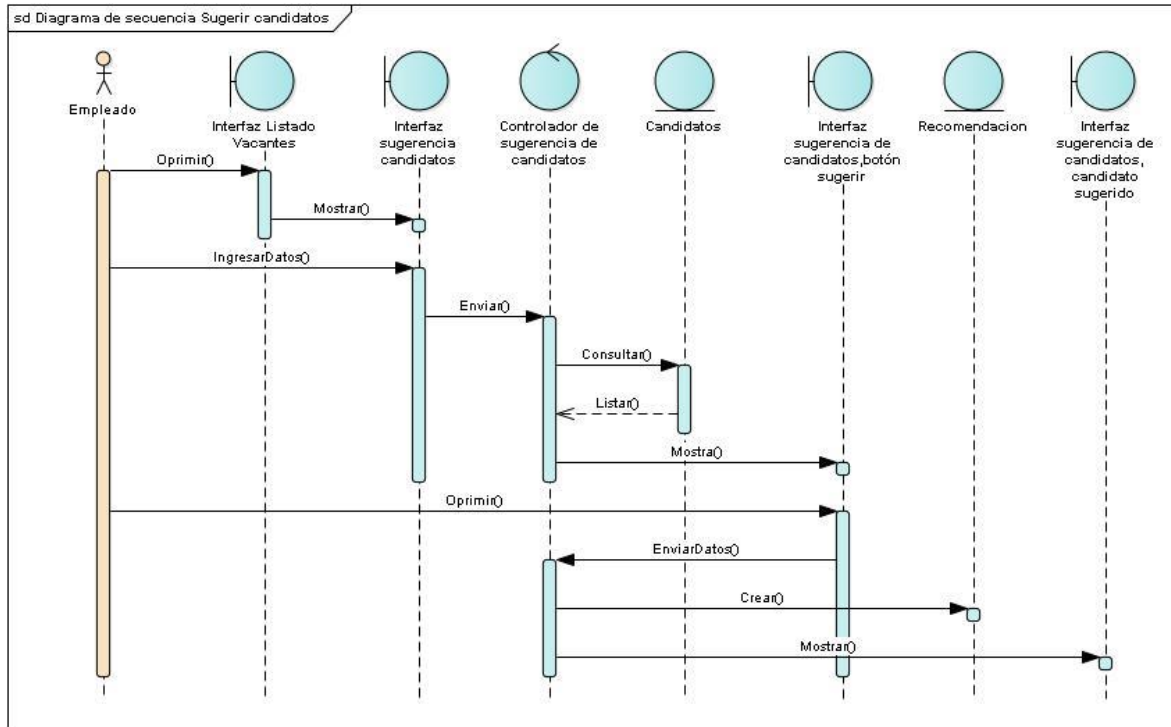
Tabla 2. Recomendar sugerencias

Nombre del caso de uso	Recomendar Sugerencias
Actor Participante	Iniciado por el Empleado
Condición inicial	1. El usuario oprime en el nombre de la vacante
Flujo de eventos	2. Se muestra la vacante con el listado de candidatos sugeridos 3. Oprime el botón de recomendar
Condición de salida	4. Muestra mensaje de candidatos enviados correctamente

Fuente: Elaboración propia

3.2.3 Caso de uso sugerir candidatos.

Figura 9. Diagrama de secuencia sugerir candidatos



Fuente: Elaboración propia

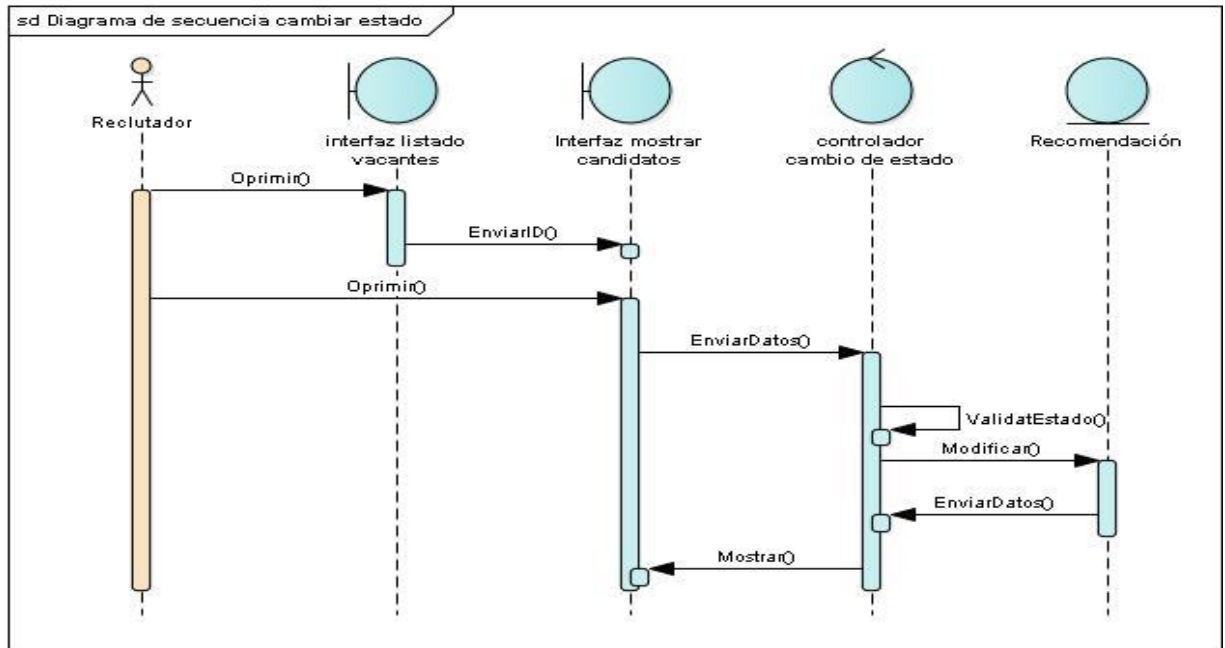
Tabla 3. Sugerir candidatos

Nombre del caso de uso	Sugerir Candidatos
Actor Participante	Iniciado por el Empleado
Condición inicial	1. El usuario oprime en el nombre de la vacante
Flujo de eventos	2. Se muestra la vacante con el botón para añadir más candidatos 3. El usuario ingresa el nombre del candidato a sugerir 4. Se despliegan los resultados de las coincidencias 5. El usuario oprime el botón de sugerir candidato
Condición de salida	6. Muestra la pantalla con el candidato agregado a “Candidatos sugeridos”

Fuente: Elaboración propia

3.2.4 Caso de uso cambiar estado

Figura 10. Diagrama de secuencia cambiar estado



Fuente: Elaboración propia

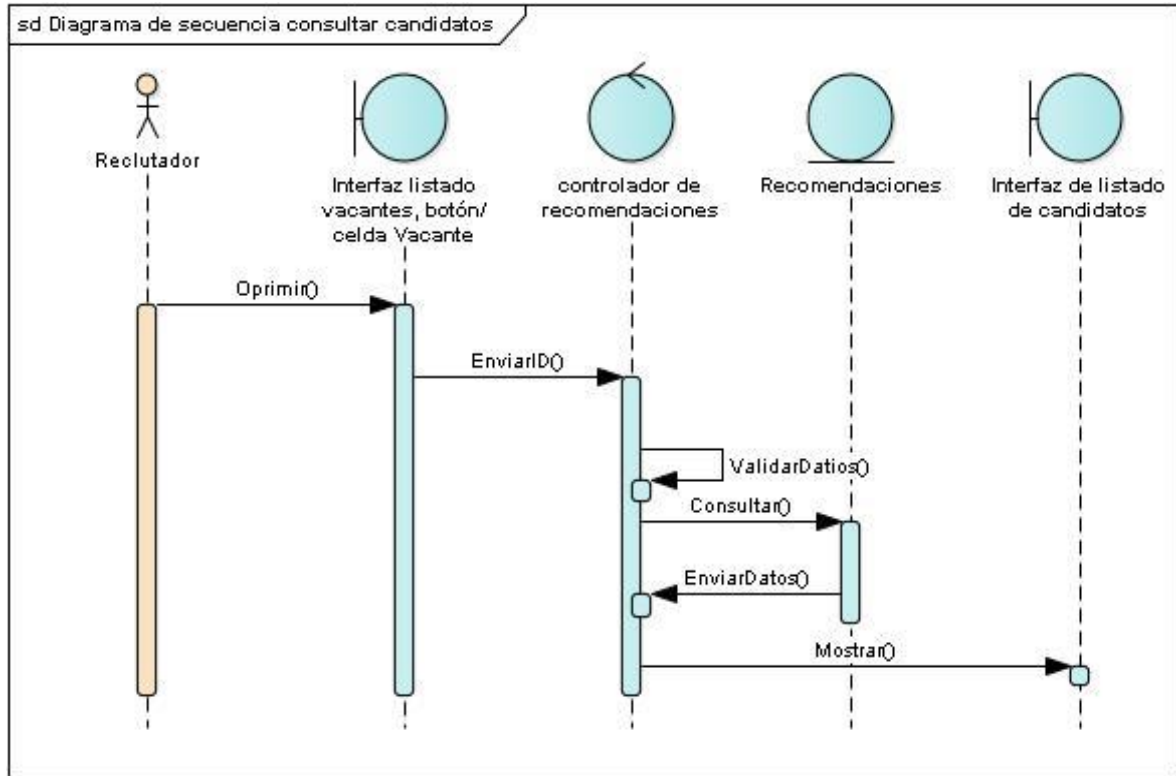
Tabla 4. Cambiar estado

Nombre del caso de uso	Cambiar Estado
Actor Participante	Iniciado por el Reclutador
Condición inicial	1. El usuario oprime en el nombre de la vacante
Flujo de eventos	2. Se muestra la vacante con el botón para añadir más candidatos. 3. El usuario presiona el botón de agregar a favoritos 4. Se traslada el candidato a la sección de Favoritos
Condición de salida	5. Muestra los usuarios en Favoritos
Requerimientos especiales	1. Si el usuario presiona en el botón de WhatsApp se irá a la columna de Contactado 2. Si el candidato está en la columna el reclutador puede arrastrar al candidato hacia donde desee (cambio de estado)

Fuente: Elaboración propia

3.2.5 Caso de uso consultar candidatos

Figura 11. Diagrama de secuencia consultar candidatos



Fuente: Elaboración propia

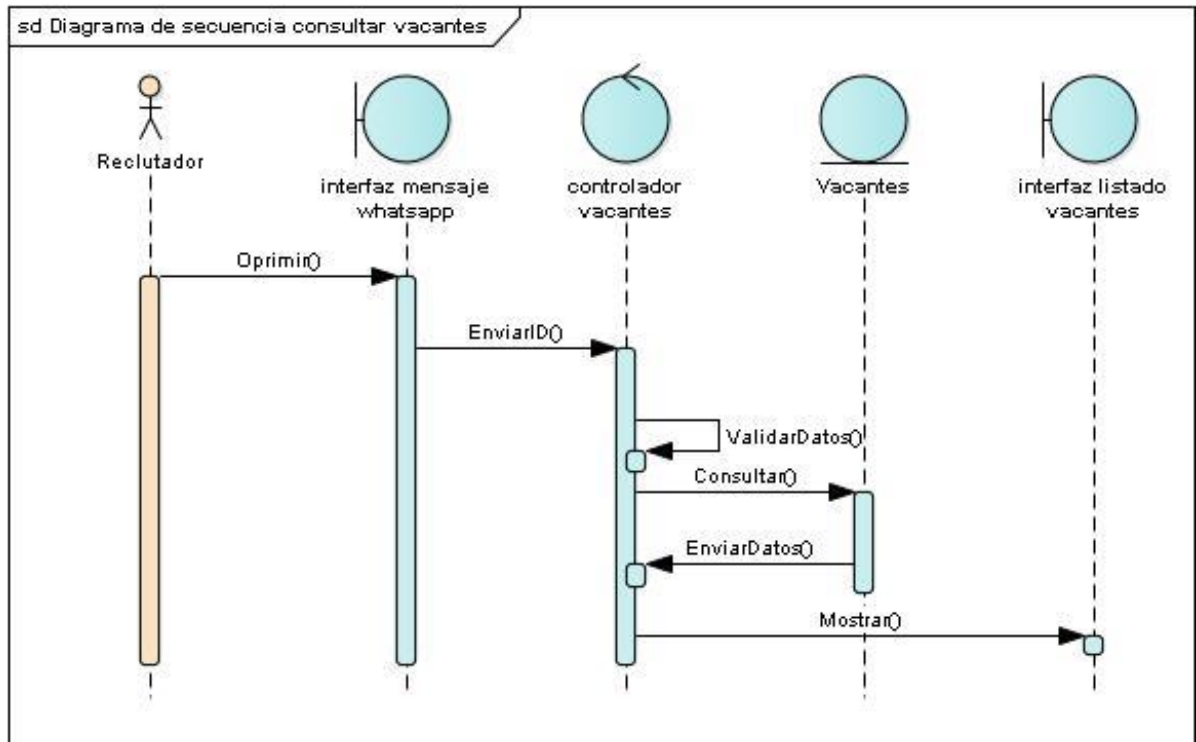
Tabla 5. Consultar candidatos

Nombre del caso de uso	Consultar Candidatos
Actor Participante	Iniciado por el Reclutador
Condición inicial	1. El usuario visualiza las vacantes
Flujo de eventos	2. El usuario oprime sobre el nombre de la vacante
Condición de salida	3. Se despliegan todos los candidatos que han sido recomendados para dicha vacante.
Requerimientos especiales	1. Si la vacante no tiene candidatos no abrirá la página los candidatos.

Fuente: Elaboración propia

3.2.6 Caso de uso consultar vacantes.

Figura 12. Diagrama de secuencia consultar vacantes



Fuente: Elaboración propia

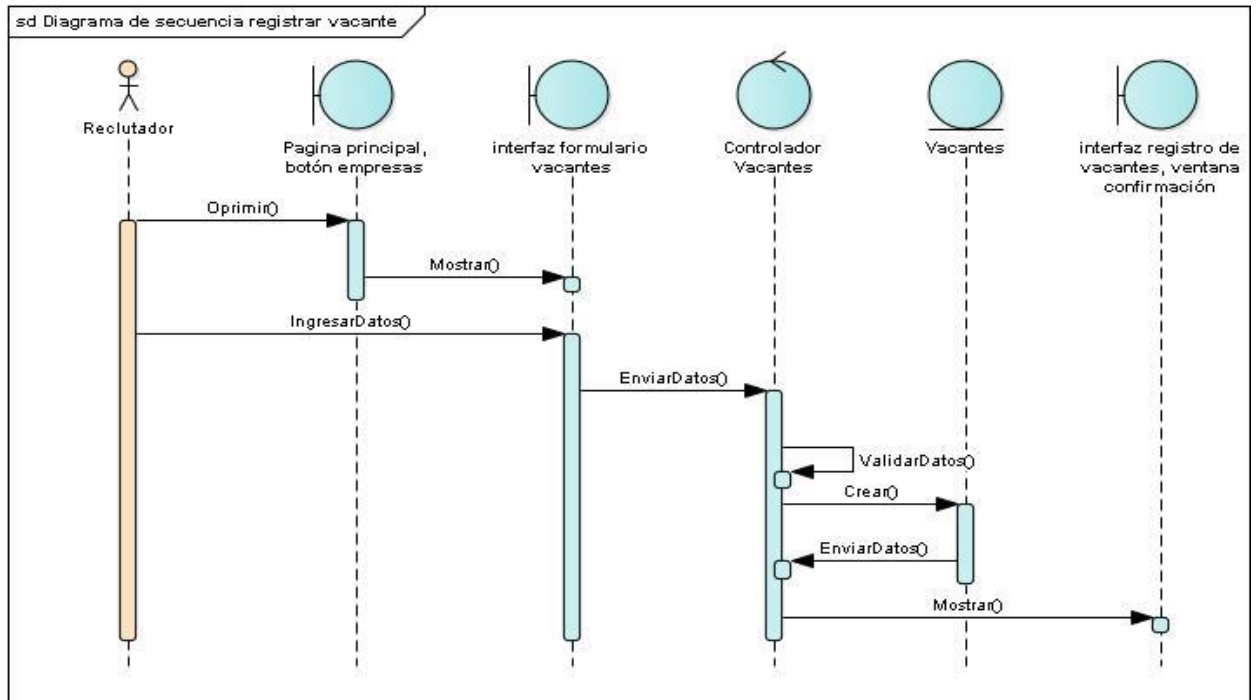
Tabla 6. Consultar vacantes

Nombre del caso de uso	Consultar Vacantes
Actor Participante	Iniciado por el Reclutador
Condición inicial	4. El usuario debe registrar vacantes
Flujo de eventos	5. El usuario oprime en el link enviado
Condición de salida	6. Se despliegan toda la información actual de las vacantes registradas

Fuente: Elaboración propia

3.2.7 Caso de uso registrar vacantes

Figura 13. Diagrama de secuencia registrar vacantes



Fuente: Elaboración propia

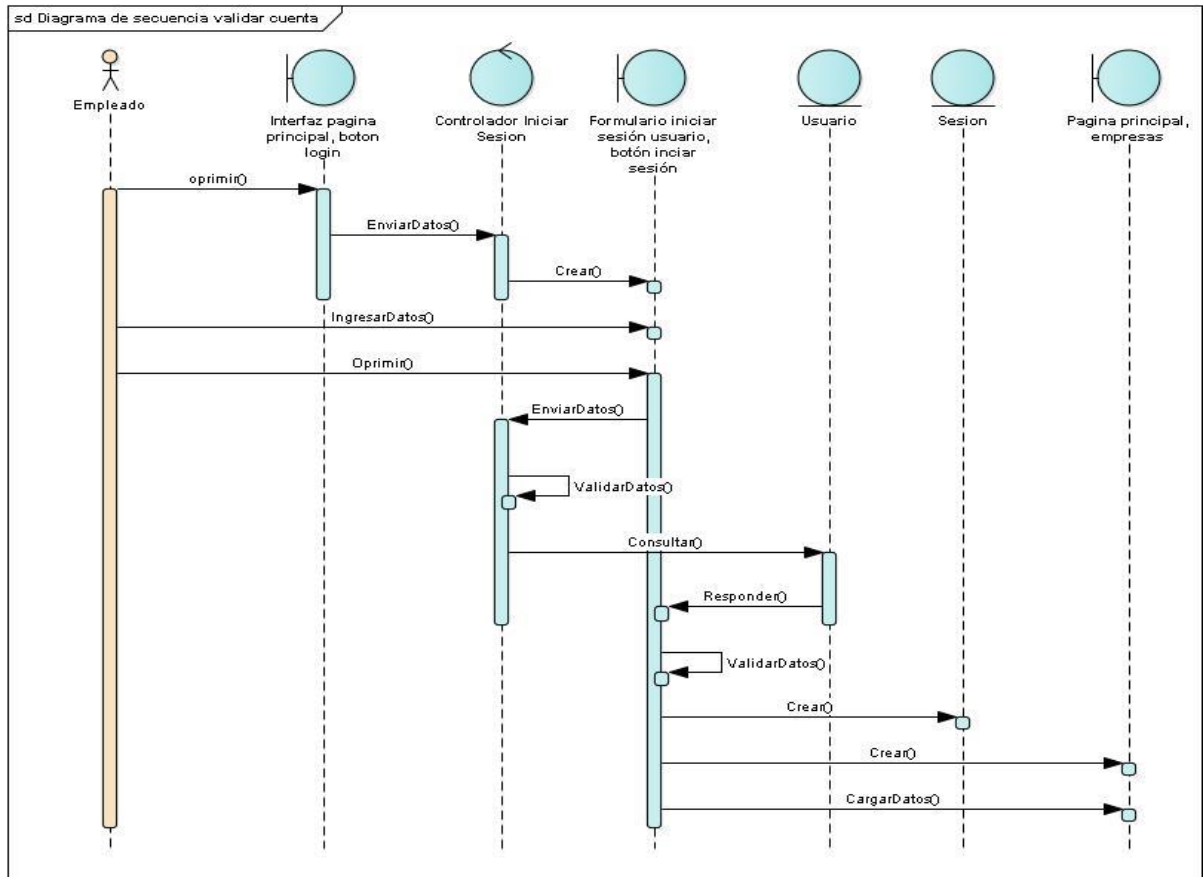
Tabla 7. Registrar vacantes

Nombre del caso de uso	Registrar Vacantes
Actor Participante	Iniciado por el Reclutador
Condición inicial	1. El usuario debe ingresar desde la página principal
Flujo de eventos	2. El usuario oprime en el botón “Para Empresas” 3. Se despliega la pantalla del formulario de vacantes 4. El usuario registra todos los campos solicitados 5. El usuario oprime en el botón “Registrar Vacante”
Condición de salida	6. Se despliega una pantalla de confirmación exitosa del registro de la vacante
Requerimientos especiales	7. Si falta algún campo por llenar, será señalado y no permitirá culminar el registro de la vacante

Fuente: Elaboración propia

3.2.8 Caso de uso validar cuenta.

Figura 14. Diagrama de secuencia validar cuenta



Fuente: Elaboración propia

Tabla 8. Validar cuenta

Nombre del caso de uso	Validar Cuenta
Actor Participante	Iniciado por el Empleado
Condición inicial	1. El usuario debe ingresar desde la página principal
Flujo de eventos	2. Ingresa el correo y contraseña 3. El usuario oprime el botón de ingresar
Condición de salida	4. Se despliega la página con las funcionalidades
Requerimientos especiales	5. Si el correo electrónico está mal escrito, saldrá un mensaje de correo electrónico mal escrito. 6. Si la contraseña está mal escrita, saldrá un mensaje de contraseña mal escrita.

Fuente: Elaboración propia

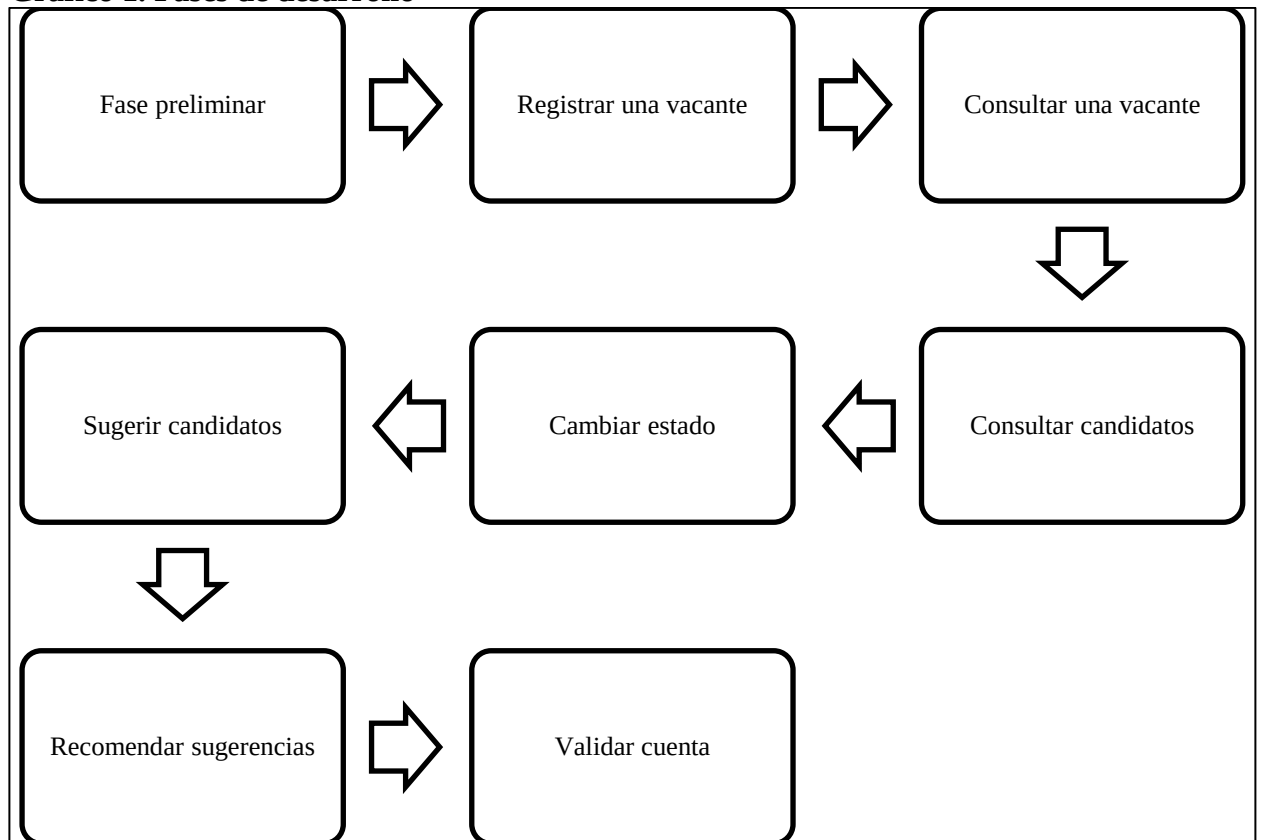
3.3 DESCRIPCIÓN DEL DESARROLLO

Para la descripción del desarrollo se procedió a exponer como se realizó el proyecto de manera general, y para cada caso de uso, teniendo en cuenta que el presente proyecto son funcionalidades web No-Code orientadas a la administración de vacantes para una agencia de empleo, donde existen otros módulos y funcionalidades que corresponden a otras áreas de la empresa que componen el negocio de una agencia de empleo.

3.3.1 Fases del desarrollo

El desarrollo web No-Code orientado a la administración de vacantes para una agencia de empleo, se divide en nueve etapas fundamentales, estas etapas son los distintos desarrollos, donde secuencialmente un desarrollo es dependiente de que el otro esté construido, por eso el orden se indica en la siguiente gráfica:

Grafico 1. Fases de desarrollo



Fuente: Elaboración propia

3.3.2 Fase preliminar

Descripción: En esta fase preliminar se define la manera general de un desarrollo que mantendrá un estándar para los procesos de cada caso de uso, en esta fase del desarrollo se describe el porqué de cada funcionalidad y su importancia para ser usada en las fases siguientes de los desarrollos.

- Inicialmente desde el área de producto era descrito con detalle el desarrollo para llevar a cabo cada funcionalidad del caso de uso, este era expuesto en la herramienta para asignación de tareas llamada “clickup”, herramienta que usa la metodología Kanban, allí eran descritas las distintas necesidades de la tarea, se dejaban detalles del problema (Pendiente), una perspectiva de solución en curso (En curso) y lo que se solucionaba finalmente (Hecho). Generalmente se dejaba adjunto un diseño que había sido suministrado por el respectivo equipo de diseño.
- Como segundo paso se procedía a observar las características de cada diseño, tales como las medidas, colores, fuentes, bordes entre otros que posteriormente eran aplicados al editor de diseño.
- Posteriormente, se accede a la página Bubble.io donde se abría el editor de Bubble, en este punto el editor se divide en siete secciones como se expone a continuación

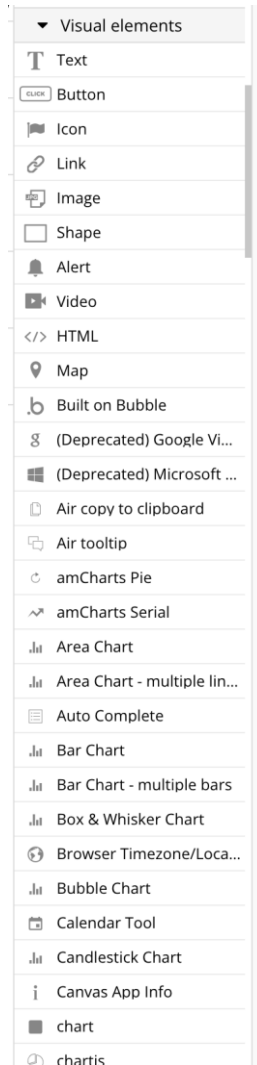
Figura 15. Editor Bubble



Fuente: (Editor Bubble)

La sección de diseño (Desing), es el editor principal donde se administran los elementos visuales que ayudarán a construir el Front-end de la página, desde allí básicamente se arrastran elementos visuales y se construyen las distintas estructuras de la plataforma para la administración de vacantes.

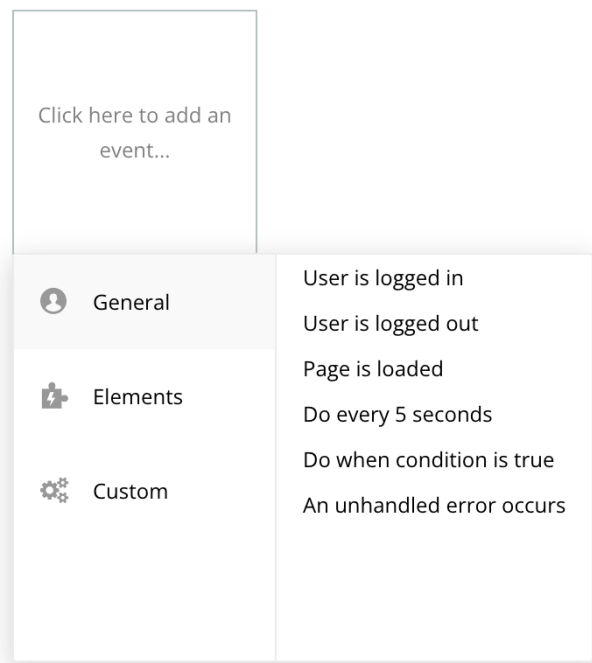
Figura 16. Sección de Desing



Fuente: (Editor Bubble)

La sección de Workflow²², es el cerebro, donde se implementa todas acciones que conectarán con la Data dependiendo a lo que se realice en los elementos visuales, por ejemplo, al accionar un botón que guarde los datos recolectados en cajas de texto.

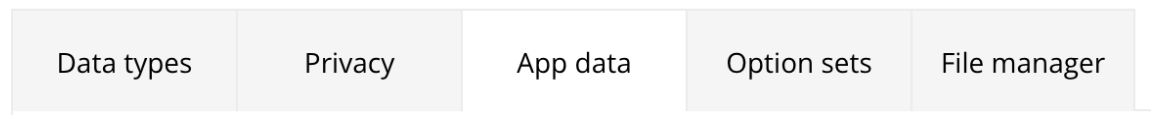
Figura 17. Sección workflow



Fuente: (Editor Bubble)

La sección de Data, en esta parte es donde se administra todos los datos, se crea la estructura de las bases de datos, también se podrá buscar, crear, modificar o eliminar registros de forma interna y manual. De igual manera crear datos estáticos que podrán optimizar el rendimiento de nuestra plataforma

Figura 18. Sección data

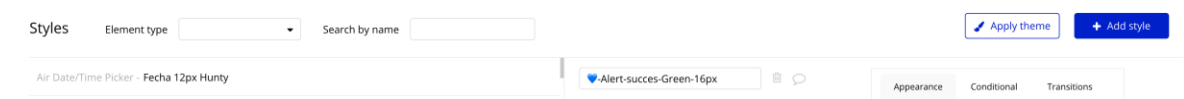


Fuente: (Editor Bubble)

²² Conjunto de actividades relacionadas, que son completadas en un determinado orden para alcanzar un objetivo de la organización

La sección de Styles, en esta sección se podría decir que es un atajo para optimizar y hacer el trabajo más fácil, aquí se puede definir de forma predefinida, estilos de letras, bordes, colores, de los distintos elementos visuales (aunque se pueden individualizar por elemento y hacer de forma manual desde el editor de diseño).

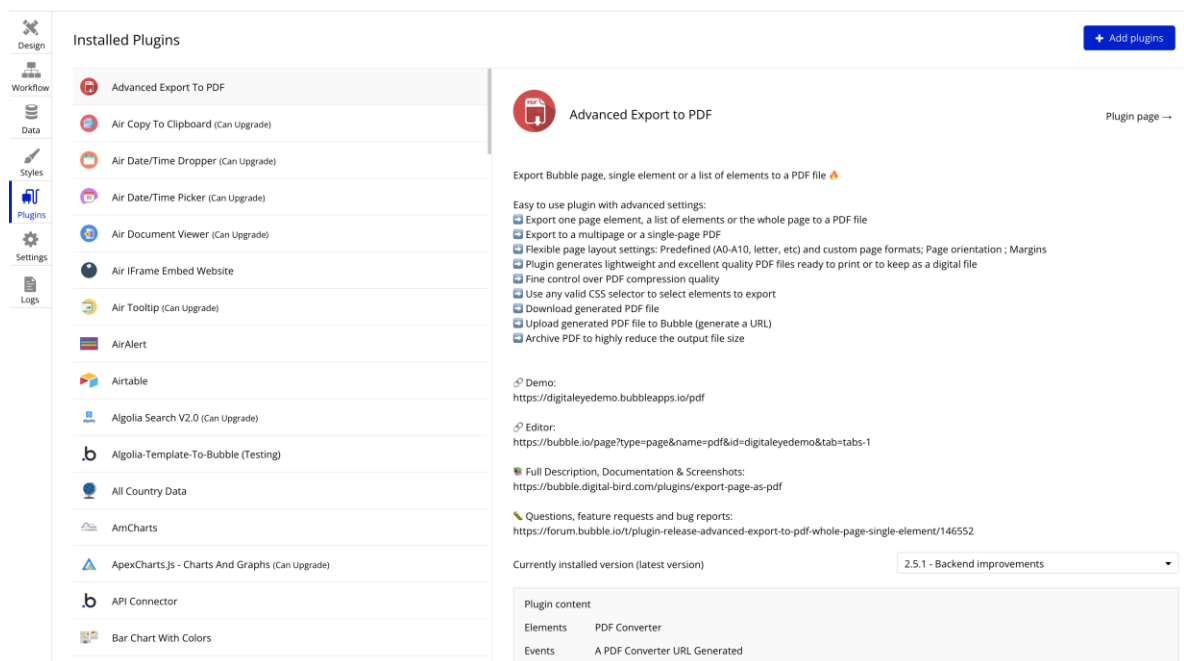
Figura 19. Sección styles



Fuente: (Editor Bubble)

La sección de Plugins, aquí básicamente se tiene una ayuda extra desde otros desarrolladores, son extensiones o agregados que prestan una ayuda en elementos especiales que cumplen una función visual o realiza alguna función especial, que no posee de manera nativa la herramienta Bubble, por ejemplo, un selector de calendario que permite realizar una selección de varios días de forma personalizada.

Figura 20. Sección de plugins



Fuente: (Editor Bubble)

La sección de Settings, aquí es donde se puede realizar ajustes a toda la plataforma de manera general, en este tipo de ajustes se puede ver las suscripciones que se han realizado por plugins, dominios, entre otros.

Figura 21. Sección de settings

App plan	General	Domain / email	Languages	SEO / metatags	API	Collaboration	Sub apps	Versions
----------	---------	----------------	-----------	----------------	-----	---------------	----------	----------

Fuente: Elaboración propia

Sección de Logs, aquí es donde se observan los movimientos y registro de eventos que ocurren en toda la plataforma, da una mano si hay alguna acción que se está ejecutando de forma errónea.

Figura 22. Sección de logs

Capacity	Server logs	Scheduler
----------	-------------	-----------

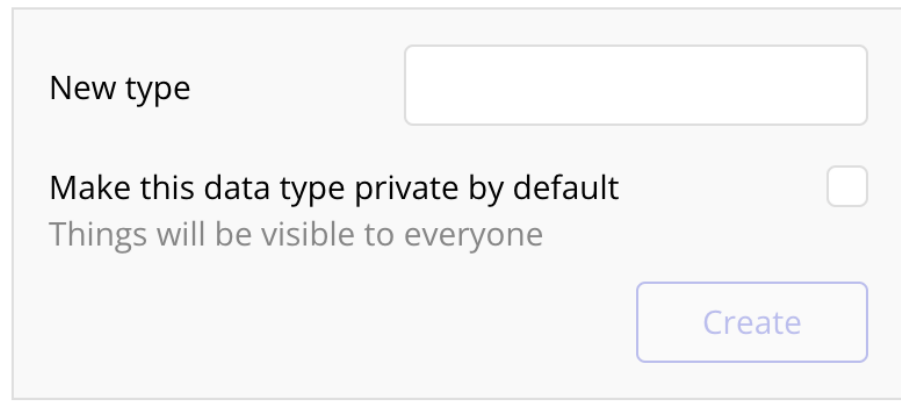
Fuente: (Editor Bubble)

3.3.3 Registrar Vacante

Descripción: En esta segunda fase se procede a realizar la construcción de la funcionalidad para registrar una vacante, es de vital importancia esta fase porque una vez constituida la forma en la que llega la información de las vacantes, se puede proceder con el siguiente desarrollo, que es consultar una vacante.

- Inicialmente se observa en una sesión con el diseñador gráfico, cómo se muestran que los elementos sean creados, y a su vez se define la estructura de datos, la cual se documenta y se discute para su creación.
- Posteriormente se procede a crear la estructura de datos con la herramienta, creando una tabla de tipo vacante, y sus respectivos campos relevantes, como nombre de la vacante, descripción, rango salarial, habilidades necesarias, ubicación modalidad, e información de contacto del reclutador.

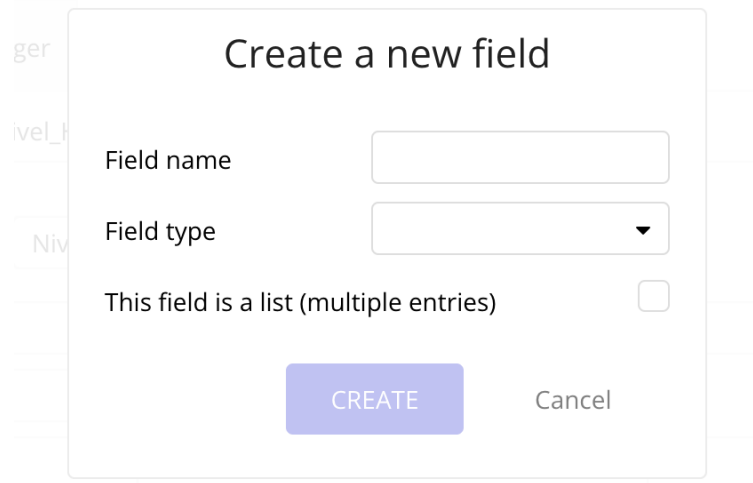
Figura 23. Creador de tablas

A light gray rectangular form with a white border. At the top, the text 'New type' is followed by a white input field. Below this, the text 'Make this data type private by default' is followed by an unchecked checkbox. Underneath the checkbox, the text 'Things will be visible to everyone' is displayed in a lighter gray font. At the bottom right, there is a blue button with the word 'Create' in white text.

Fuente: (Editor Bubble)

Para crear la tabla se dirige a la sección principal de “Data” y se registra el nombre de la tabla y se oprime en “Create”

Figura 24. Creador de campos

A light gray rectangular form with a white border. At the top, the title 'Create a new field' is centered. Below the title, the text 'Field name' is followed by a white input field. Underneath, the text 'Field type' is followed by a white dropdown menu with a downward arrow. Below the dropdown, the text 'This field is a list (multiple entries)' is followed by an unchecked checkbox. At the bottom, there are two buttons: a blue button with the word 'CREATE' in white text, and a gray button with the word 'Cancel' in gray text.

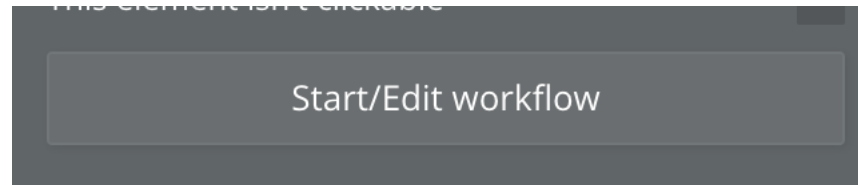
Fuente: (Editor Bubble)

De igual forma para crear los campos, se especifica el nombre del campo y su tipo (numero, texto, imagen, booleano, etc.)

- Una vez realizada la creación de la estructura, se realiza el desarrollo de la estructura gráfica o Frontend, para esto, se arrastran los elementos necesarios, tales como cajas de texto, textos, imágenes y botones, que harán visualmente disponible la página para su uso.

- Una vez disponibles los elementos gráficos, se procede a realizar las acciones o Workflows que guardarán nuestra información en la Base de Datos, para esto, el botón de Enviar Vacante será el principal implicado en la configuración de esta funcionalidad.

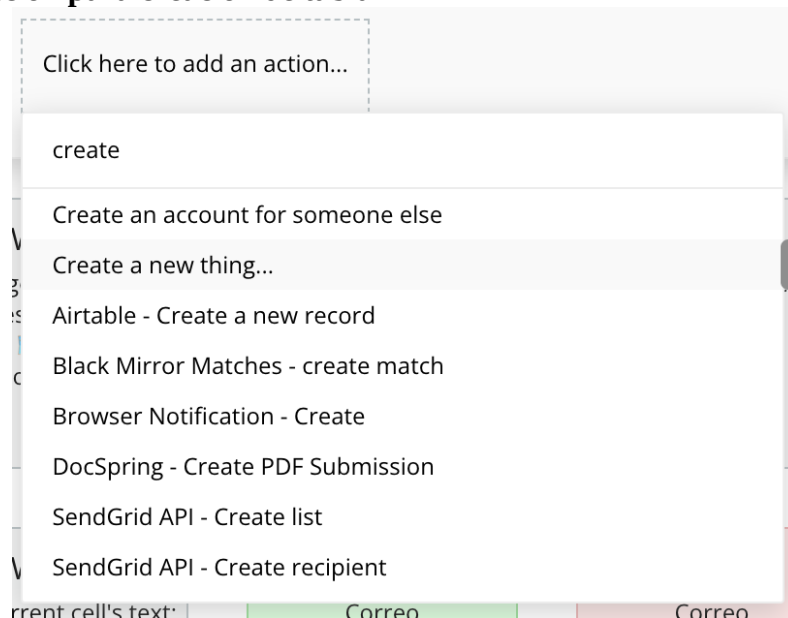
Figura 25. Botón para Workflow



Fuente: (Editor Bubble)

Dentro de las opciones que ofrece el botón “Enviar vacante” se encuentra la opción para iniciar una acción o Workflow, esta opción nos redirige a la segunda sección del desarrollo general descrito.

Figura 26. Acción para creación de tabla



Fuente: (Editor Bubble)

- Se creo una nueva opción, y se selecciona “Create a new thing”, dentro de esta opción se desplegará y se preguntará que se quiere realizar la creación, en este caso, se creará una nueva vacante, y así mismo se permite rellenar los datos completados en el formulario, para esto se pregunta también que campos de la tabla se quiere llenar, y es allí cuando se selecciona uno a uno y se le indica que caja de texto o elemento es el que lo llena, finalmente queda guardada esta edición.

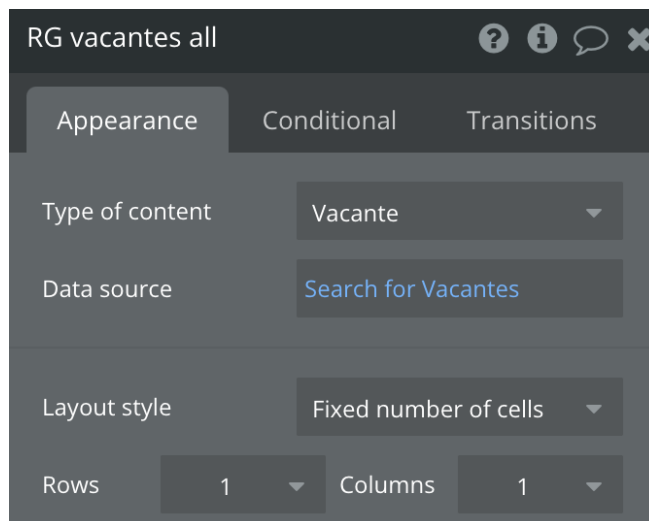
3.3.4 Consultar una vacante

Descripción: En esta tercera fase se procede a realizar la funcionalidad para consultar una vacante, es en este punto se describe como se construye la funcionalidad que listará las vacantes del reclutador.

En este punto, es fundamental recordar que Bubble no posee una estructura Backend explícita para el usuario, hablando en términos de lo que se ve como desarrollador en Bubble, por tanto, ya todo está demasiado intuitivo y resumido para consumir desde el Frontend. El desarrollo a continuación, es realmente sencillo de describir.

- De vuelta al editor gráfico, existe un elemento que se llama “Repeating group”, este elemento es un elemento que lista visualmente lo contenido en una tabla, dentro de este elemento se deben arrastrar los elementos que se quieren mostrar, textos, imágenes, videos, o incluso botones para ejecutar acciones. En este caso se arrastra textos para mostrar información de la vacante.
- Básicamente ejecutando la opción “Do search for” (Buscar en) se selecciona la tabla que se desea que sea listada por este grupo de repetición, de esta forma se alimenta la información, se añaden condiciones de ser necesario para la búsqueda, esta condición será que solo se muestren las vacantes correspondientes al reclutador, y de esta forma se alimenta lo que se va a mostrar en al reclutador, en este caso, vacantes.

Figura 27. Configuración grupo de repetición



Fuente: (Editor Bubble)

3.3.5 Consultar candidatos

Descripción: En esta cuarta fase, se procede describir la construcción de la funcionalidad para la plataforma donde el reclutador podrá consultar los candidatos que le han sido enviados como recomendaciones.

Para describir la construcción de dicha funcionalidad realmente es sencillo, al igual que para la construcción de la funcionalidad para consultar vacantes, solo basta con incluir un grupo de repetición para listar los candidatos de la vacante, repitiendo los pasos de la construcción anterior con la diferencia que el Tipo de Contenido serán Candidatos y el Data Source (fuente de datos) será un Search for (Consulta) Candidatos.

3.3.6 Cambiar estado

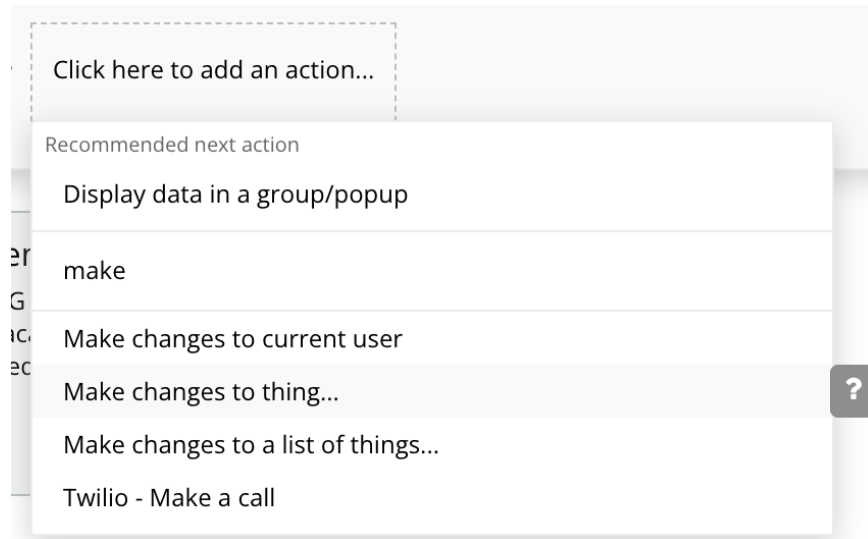
Descripción: En esta quinta fase, se procede a describir el desarrollo para realizar la construcción de la funcionalidad para cambiar el estado de una recomendación, este desarrollo se realiza para que el reclutador pueda establecer el estado actual del candidato en la vacante.

Esta fase se basa en la fase del desarrollo para consultar candidatos, para esta construcción, se agrega al diseño creado que se repite, dos botones, uno para contactar al candidato y otro para agregar a favoritos.

El botón para contactar al candidato, básicamente ejecuta un script que abre una ventana de WhatsApp, pero el punto más importante aquí es que una vez esto se ejecuta, se cambia el estado de la recomendación del candidato, ahora guardará en la base de datos, que fue contactado, para esto se selecciona la opción Make changes (realizar cambios) para un solo objeto, y dentro de esto, se indica específicamente por medio de un Search For, el candidato al cual será cambiado el estado a Contactado.

Por su parte, el desarrollo para agregar a Favoritos se no realiza ninguna acción externa, sino que simplemente cambiar el estado de la recomendación del Worki a Favorito también por una modificación al objeto.

Figura 28. Acción para modificar una tabla



Fuente: (Editor Bubble)

3.3.7 Sugerir candidatos

Descripción: En esta sexta fase, se describe la construcción de la funcionalidad para sugerir candidatos, este se lleva a cabo para que en la vista del empleado pueda dar a conocer las sugerencias de los candidatos a recomendar.

En esta fase del desarrollo inicialmente se debió implementar una estructura visual para listar los candidatos, pero con una variación particular, hay que condicionar este grupo de repetición a una caja de texto, esto con el fin de que solo muestre los resultados que tengan coincidencias con lo digitado en la caja de texto, esto se realizó añadiendo una condición a la condición del “Search For Candidato” (Buscar en la tabla Candidato) indicándole que el nombre del candidato debe ser igual al de la caja de texto, a partir de allí únicamente mostrará los resultados digitados.

Una vez construido el grupo de repetición, se procedió a desarrollarlo funcionalmente, en este punto se agrega un botón para que en cada resultado, si es presionado se cree un elemento de tipo recomendación, que contiene un objeto de tipo vacante, otro de tipo candidato y un texto que indica el estado de la recomendación, si, este mismo que se explica en desarrollos anteriores, este será de tipo Sugerencia, todo esto se hace con el mismo desarrollo que se usa para el registro de una vacante, la única variación esta vez serán los campos de una recomendación.

3.3.8 Recomendar sugerencias

Descripción: En esta séptima fase, se describe la construcción de la funcionalidad para recomendar sugerencias, en este punto se construye el cambio de estado para las sugerencias resultantes del desarrollo anterior.

Para esta fase del desarrollo se arrastra un grupo de repetición al editor gráfico que tendrá como fuente una Búsqueda de Recomendaciones.

Una vez esto realizado, se arrastra un botón a esta celda (es lo que tendrán cada grupo de repetición), a este botón hace que se modifique la recomendación, con un Make Changes, se cambia el estado de la recomendación a “Recomendado”.

3.3.9 Enviar vacantes

Descripción: En esta octava fase, se describe la construcción del envío de vacantes que realizan los trabajadores internos con las respectivas recomendaciones (candidatos) hacia los reclutadores de las distintas empresas aliadas, quienes han registrado previamente sus vacantes usando el desarrollo anterior.

Inicialmente al igual que en la fase para consultar vacantes, se construye un grupo de repetición, pero de cara al empleado de la empresa, el cual será el encargado de enviar las vacantes una vez hayan sido revisadas, ya sea porque tengan o no vacantes.

Se añade un botón para la casilla a repetir, y en este se añade la acción del Script para enviar un mensaje de WhatsApp, en él, contiene texto que a su vez contiene un vínculo, este vínculo es la página que abrirá en Reclutador para ver su listado de vacantes, es decir el desarrollo que se describe para Consultar Vacantes.

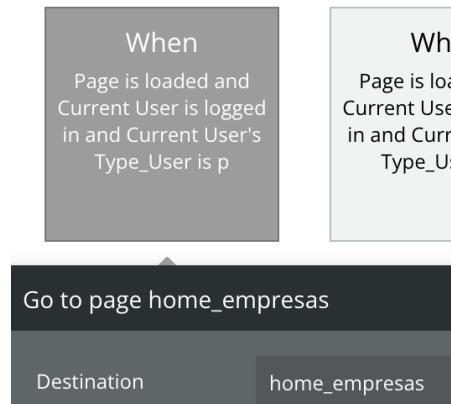
3.3.10 Validar cuenta

Descripción: En esta novena fase, se describe la construcción de la funcionalidad que realiza las validaciones de los usuarios, implementando una validación para el tipo de usuario, ya sea empleado o reclutador.

El alcance de este proyecto está descrito únicamente llega hasta la administración de vacantes para una agencia de empleo, es decir, existen otras funcionalidades encargadas de otros puntos en la empresa, previamente, de forma paralela se han creado funcionalidades encargadas del registro e inicio de sesión del usuario.

No obstante, una vez validada la cuenta, es necesario dentro de esta validación condicionar el tipo de usuario que inicie sesión.

Figura 29. Workflow para validar un tipo de usuario



Fuente: (Editor Bubble)

Para el caso de esta funcionalidad, se ejecutó un paso que tiene descrito Bubble como “Go to page” (Ir a la página), el cual se encargará de enviar a esta funcionalidad, y dependiendo el tipo de usuario que sea clasificado, si se es un empleado, irá al home de empresas, y si se es un reclutador, irá al listado de vacantes que se describió en el desarrollo para consultar vacantes.

3.4 VALIDACIÓN DEL DESARROLLO

En este capítulo se explicará cómo se realizó la validación del prototipo que se implementó en la empresa donde se realizaron las respectivas prácticas y cada uno de los desarrollos para cada caso de uso. Para esto es importante recordar que existen dos actores o tipos de usuario principales, el reclutador y el empleado. Las distintas validaciones de los desarrollos y el prototipo no siguieron un estándar basado en alguna metodología por lo que se tendrá en cuenta solo cuatro elementos fundamentales, dos en la etapa final del desarrollo: la validación funcional del caso de uso, y la validación visual del diseño; dos correspondientes a los actores o usuarios finales que dan uso a la plataforma: validación de rendimiento y validación de usabilidad del producto. Es importante resaltar que las validaciones de rendimiento se realizaron a través de la herramienta de Google PageSpeed Insights²³

3.4.1 Validación en un prototipo

Para validar el desarrollo, el prototipo se configuró de la siguiente manera:

²³ Es un conjunto de herramientas de Google Inc, diseñada para ayudar a optimizar el rendimiento de un sitio web.

- Dentro del montaje del prototipo, se realizó la configuración de nombres, estilos y lenguajes de diseño propios de la empresa en la cual se implementó el prototipo y sus pruebas.
- La creación de un ambiente de desarrollo y un respectivo dominio, lo que suministró automáticamente la herramienta de desarrollo Bubble al ejecutar una visualización del producto realizado; De igual forma las respectivas rutas de dirección hacia las distintas páginas una vez previamente autorizadas, se denominaron:
 - <https://site.cazado.com/version-dev/vacantes> Registro de vacantes
 - https://site.cazado.com/version-dev/listado_vacantes Listado de vacantes
 - <https://site.cazado.com/version-dev/candidatos> Listado de recomendaciones
 - <https://site.cazado.com/version-dev/login> Validador de cuenta
 - https://site.cazado.com/version-dev/home_empresas Gestor de vacantes (Empleado)
 - https://site.cazado.com/version-dev/vacantes_candidatos Gestor de recomendaciones (Empleado)
- El prototipo fue configurado para funcionar en cualquier navegador web con soporte actualizado y/o sistema operativo, ya sea desde escritorio o móvil, característica que provee de forma automática la misma herramienta No-code.
- El prototipo se configuró entre el desarrollador, el UX/UI Designer²⁴, y uno de los empleados de la empresa.

El primer paso con respecto a la validación del prototipo, fue la creación de usuarios de prueba con rol de Empleado directamente en la base de datos, este punto se realiza ya que el desarrollo de la funcionalidad de registro no está dentro del alcance de este proyecto.

Posteriormente el empleado procedió crear vacantes de prueba donde el desarrollo fue exitoso, la validación que el mismo empleado, diseñador y desarrollador realizaron de forma detallada.

Luego el mismo empleado procedió a asignar sugerencias de recomendación a la vacante, así mismo recomendó las sugerencias con candidatos que fueron agregados a la base datos directamente, posteriormente envió dichas recomendaciones a las vacantes de prueba creadas, desarrollo que resultó exitoso.

Finalmente, en la última etapa de validación del desarrollo del prototipo, se comprobó que los enlaces de las recomendaciones fuesen enviadas al correo electrónico y WhatsApp (servicio de mensajería) previamente diligenciados en la vacante, lo cual fue exitoso, a lo que procedió a cambiar de estado las recomendaciones, marcando como favorito, descartando o contactando a los candidatos, esto con los distintos botones, en todos los casos, el desarrollo se validó de forma exitosa en el prototipo.

²⁴ Es un diseñador de interfaces de usuario para desarrollo de software.

En las validaciones siguientes se detallará el lanzamiento del prototipo con usuarios reales en cada caso de uso, validaciones que en la introducción de este capítulo fueron detalladas.

3.4.2 Validación para validar cuenta

En la primera etapa de validación del caso de uso, se llevó a cabo con una cuenta de usuario de prueba, donde inicialmente al iniciar sesión con credenciales del usuario de forma correcta se dirigió a la página principal, mientras que ingresando con las credenciales de forma errónea se denegó el ingreso y arroja correctamente un mensaje donde indica que las credenciales son incorrectas.

En la etapa de diseño, la cual fue evaluada por un UX/UI Designer, quien es la persona que realizó la creación de las distintas funcionalidades y quien para la respectiva validación tuvo en cuenta los siguientes puntos en comparación del diseño versus el resultado final, donde 1 es poco semejante, y 5 cumple con el diseño.

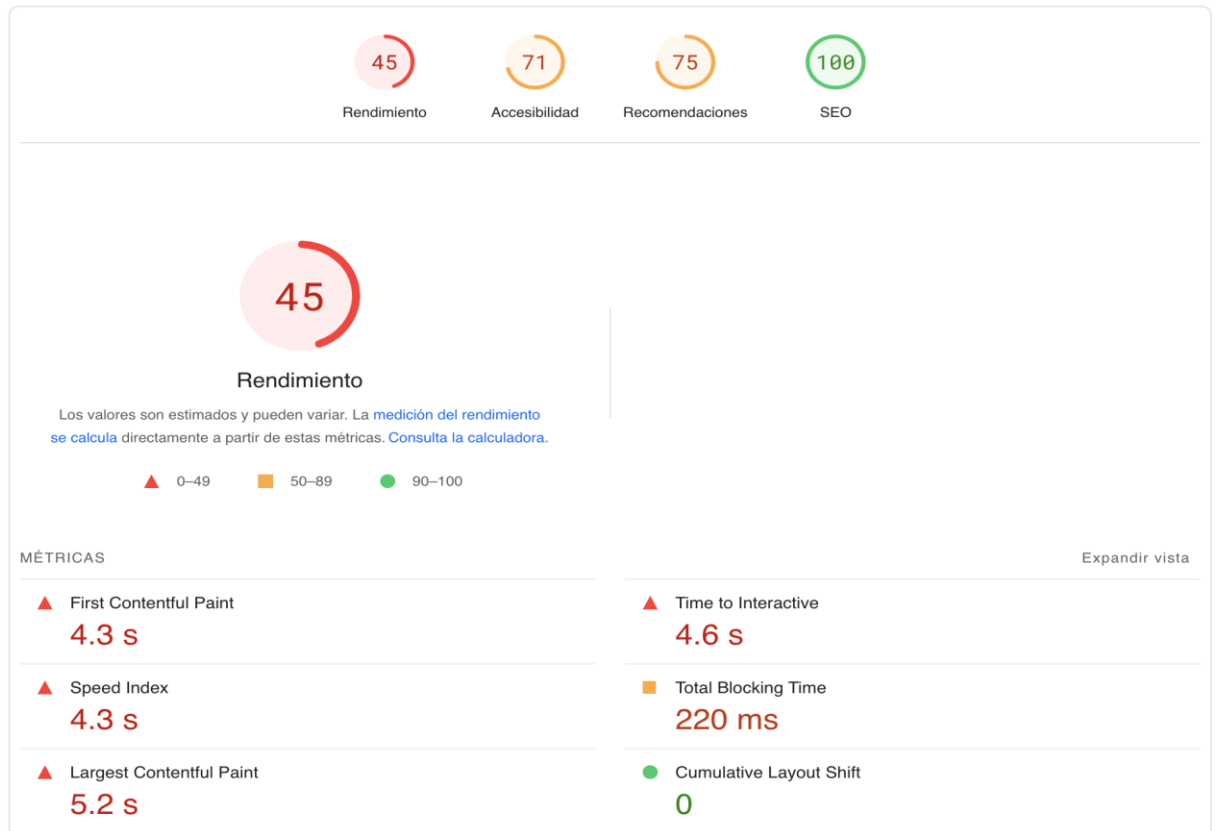
Tabla 9. Calificación de diseño validar cuenta

Características	Similitud (1-5)
Colores	5
Proporciones y dimensiones	4
Fuentes y estilos	5
Responsividad para móvil	5

Fuente: Elaboración propia.

Así mismo en esta etapa de validación con respecto al usuario final, en este caso el empleado, el cual es el único que posee una cuenta de usuario para el uso de la funcionalidad; uno de los trabajadores cuyo nombre no puede ser relevado, manifestó que: “Es fácil acceder al sitio y la experiencia es amigable para el usuario, de fácil interpretación y entendimiento para una página de inicio de sesión y validación de cuenta, sin embargo la velocidad de carga de la página puede ser levemente inferior.”.

Figura 30. Prueba de rendimiento validar cuenta



Fuente: Elaboración propia. Herramienta de Google PageSpeed Insights

Aquí se comprobó que, aunque la funcionalidad es accesible, puede tornarse un poco lenta en equipos de gama baja, aun así, es lo normal en el uso de plataformas No-code.

3.4.3 Validación para sugerir candidatos

En esta segunda etapa de validación del caso de uso, en el caso de funcionalidad se llevó se realizó con una cuenta de usuario de prueba, donde inicialmente al realizar sugerencias de candidatos a las vacantes, las búsquedas se realizan obteniendo buenos resultados, así como al agregar dichas sugerencias de recomendaciones con el botón correspondiente, la información que se comprueba en la base de datos es la que está correctamente registrada.

En la subetapa de diseño, este punto fue evaluado por un UX/UI Designer, quien tuvo en cuenta los siguientes puntos en comparación del diseño versus el resultado final, donde 1 es poco semejante, y 5 cumple con el diseño.

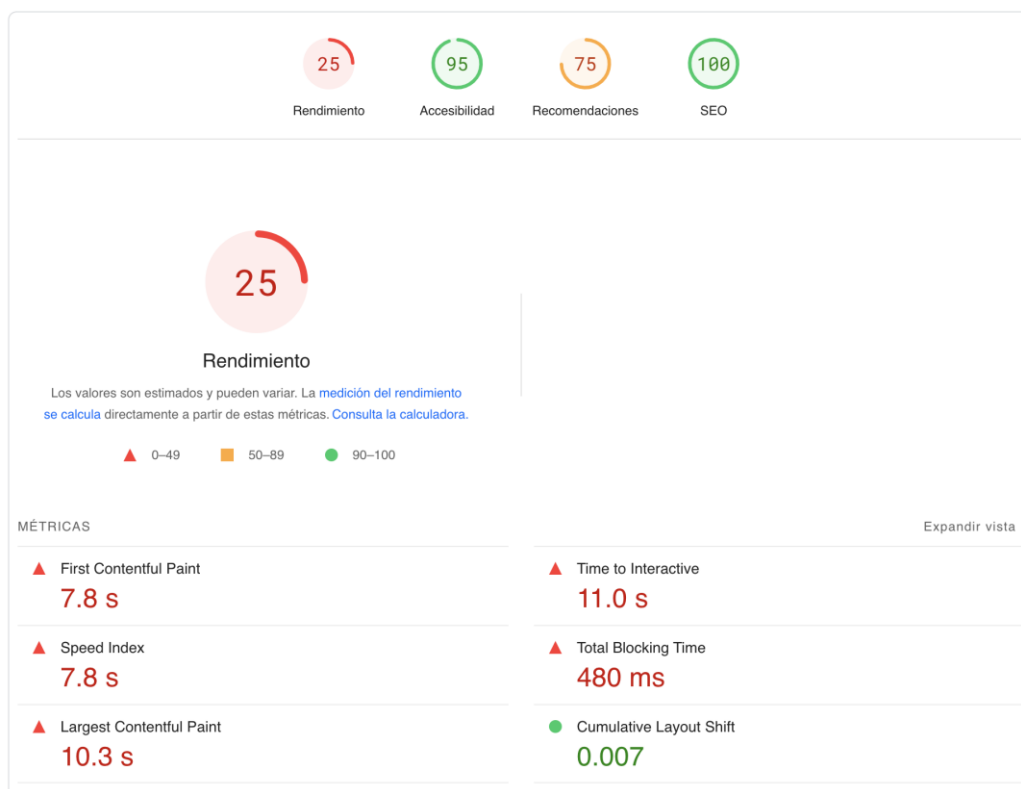
Tabla 10. Calificación de diseño sugerir candidatos

Características	Similitud (1-5)
Colores	5
Proporciones y dimensiones	5
Fuentes y estilos	5
Responsividad para móvil	5

Fuente: Elaboración propia.

En la segunda etapa de validación de para el usuario final, quien contaba con una cuenta de acceso a las funcionalidades, y por medio de uno de los trabajadores cuyo nombre no puede ser relevado, se pudo establecer que la experiencia a la hora de realizar sugerencias de candidatos es amigable y de fácil interpretación, incluso para alguien que reciba su primera inducción en esta área de la empresa, el ingreso a la página puede tornarse un poco lenta.

Figura 31. Prueba de rendimiento sugerir candidatos



Fuente: Elaboración propia. Herramienta de Google PageSpeed Insights

Se determinó la accesibilidad de la funcionalidad es positiva, pero puede tornarse un poco lenta su usabilidad en equipos de gama baja y media-baja.

3.4.4 Validación para recomendar sugerencias

Esta etapa se llevó a cabo con una cuenta de usuario de prueba, donde al recomendar las sugerencias que previamente se han realizado en otro caso de uso, visualmente se apreció que ocurre el flujo correcto donde cambia de “sugerido” a “recomendado”. Así mismo para el tema de diseño se tuvo en cuenta la evaluación de UX/UI Designer, donde se obtuvo los puntos en comparación del diseño versus el resultado final, donde 1 es poco semejante, y 5 cumple con el diseño.

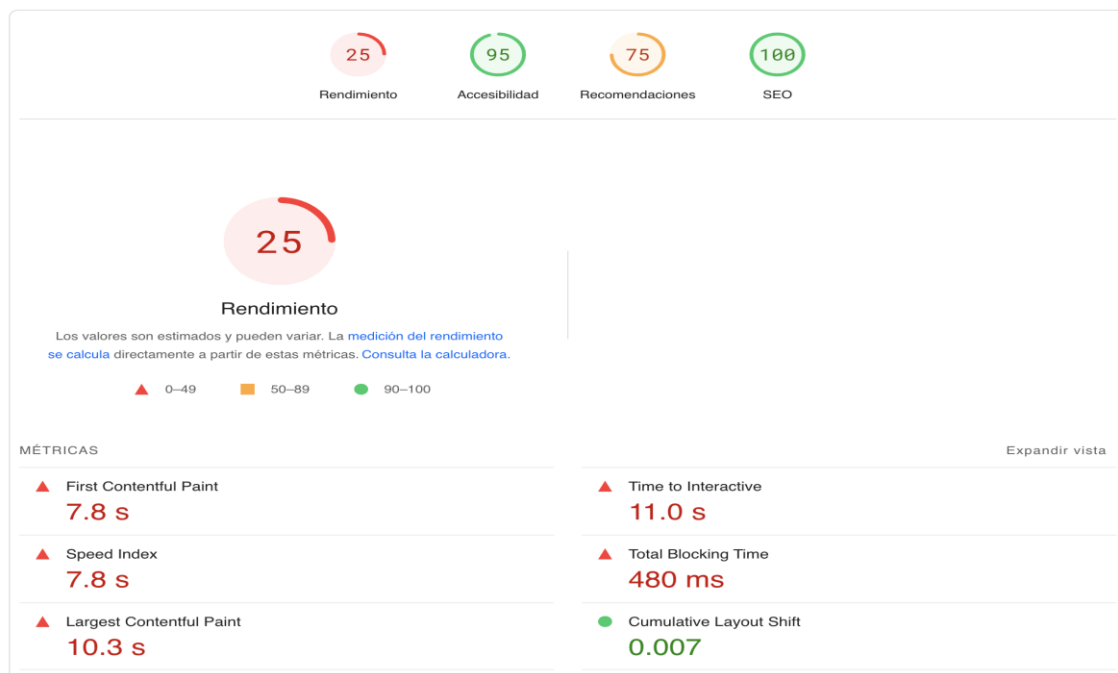
Tabla 11. Calificación de diseño recomendar sugerencias

Características	Similitud (1-5)
Colores	5
Proporciones y dimensiones	5
Fuentes y estilos	3
Responsividad para móvil	5

Fuente: Elaboración propia.

Se pudo determinar a través del usuario final, quien realizó el uso de las funcionalidades, que es muy sencillo recomendar las sugerencias que previamente se han realizado, tan fácil como oprimir un botón; la experiencia en el rendimiento de la carga de la página es la misma que la de realizar sugerencias, dado que, en efecto, es la página.

Figura 32. Prueba de rendimiento recomendar sugerencias



Fuente: Elaboración propia. Herramienta de Google PageSpeed Insights

La conclusión de la validación realizada en el desarrollo anterior fue la misma, un rendimiento de carga baja desde que se use en equipos de gama baja o media baja.

3.4.5 Validación para enviar vacantes

En este caso nuevamente se utilizó una cuenta de usuario de prueba, donde al poner a prueba la funcionalidad, se comprobó que el link con las recomendaciones realizadas haya sido enviado correctamente al correo electrónico o número de WhatsApp.

La evaluación del diseño estuvo cargo del UX/UI Designer, donde se obtuvieron unos resultados donde cuenta los siguientes puntos en comparación del diseño versus el resultado final, donde 1 es poco semejante, y 5 cumple con el diseño.

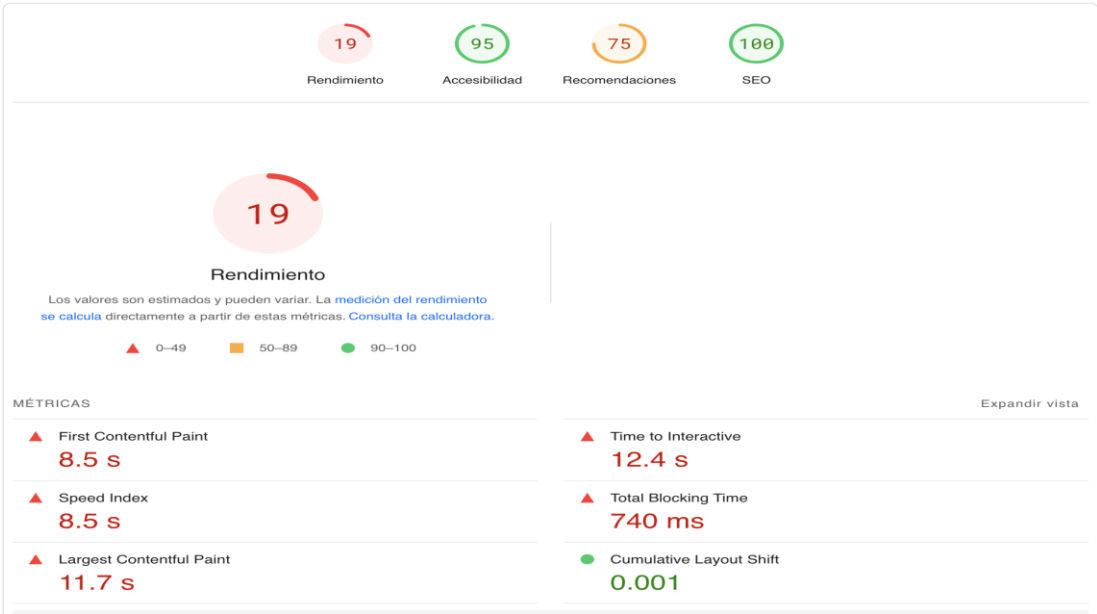
Tabla 12. Calificación de diseño enviar vacantes

Características	Similitud (1-5)
Colores	5
Proporciones y dimensiones	5
Fuentes y estilos	5
Responsividad para móvil	5

Fuente: Elaboración propia.

El usuario final expreso que la funcionalidad es una de las más sencillas de ejecutar, sin embargo, depende que el reclutador haya ingresado muy bien los datos de contacto, la experiencia por tanto es amigable y efectiva.

Figura 33. Prueba de rendimiento enviar vacantes



Fuente: Elaboración propia. Herramienta de Google PageSpeed Insights

Esta plataforma es la que más realiza consultas a la base de datos, lo cual además de los resultados anteriores, valida que su rendimiento es aún más bajo al sumar este agravante, sin embargo, sigue siendo funcional para la experiencia del empleado.

3.4.6 Validación para registrar vacante

En esta etapa se registró una vacante con todos los datos necesarios, nombre del cargo, empresa, habilidades necesarias, aspiración salarial, experiencia requerida y descripción detallada de la vacante, se probó la obligatoriedad de cada campo y los mismos se comportaron de forma correcta al no ser llenados, señalando la necesidad de ser completados correctamente.

En la subetapa de diseño, este punto fue evaluado por un UX/UI Designer, donde se lograron unos resultados teniendo en cuenta los siguientes puntos en comparación del diseño versus el resultado final, donde 1 es poco semejante, y 5 cumple con el diseño.

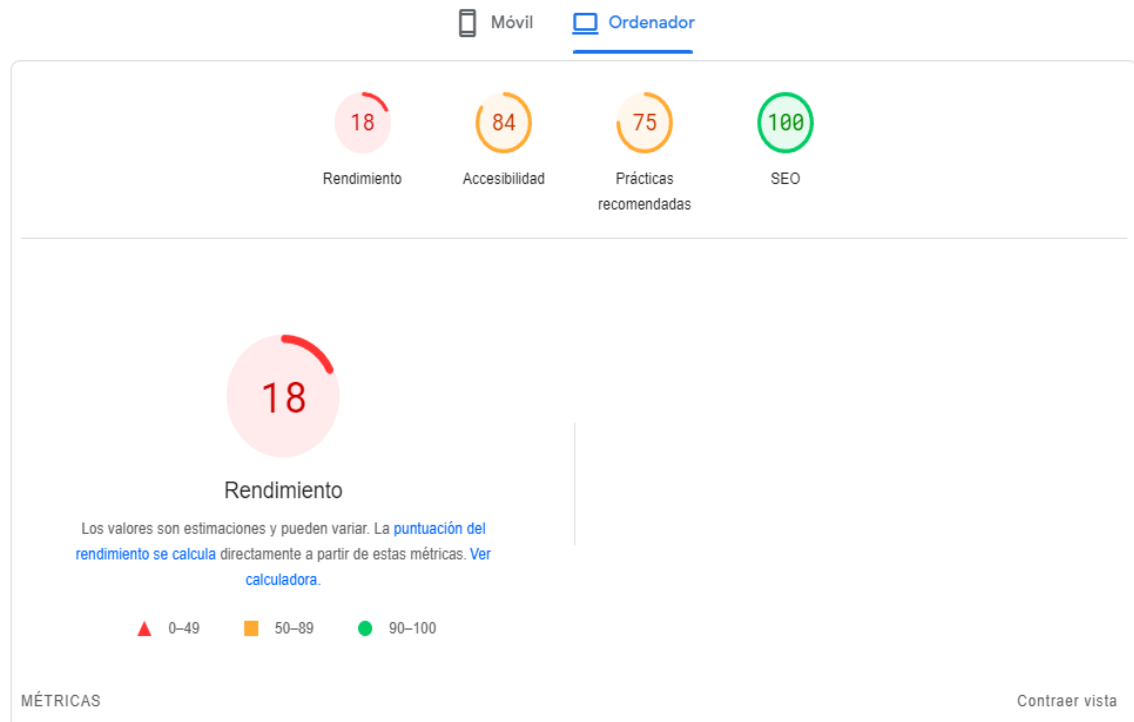
Tabla 13. Calificación de diseño registrar vacantes

Características	Similitud (1-5)
Colores	5
Proporciones y dimensiones	5
Fuentes y estilos	5
Responsividad para móvil	5

Fuente: Elaboración propia.

Como resultado el usuario final en este caso el reclutador, manifestó la facilidad al momento de llenar el requerimiento de una vacante, ya que se tienen claridad de los datos necesarios para dicho proceso además de que el comportamiento del cargado del formulario fue el ideal.

Figura 34. Prueba de rendimiento registrar vacantes



Fuente: Elaboración propia. Herramienta de Google PageSpeed Insights

Aquí se comprobó que, aunque la funcionalidad es viable, puede tornarse un poco lenta su usabilidad dependiendo del tipo de equipo utilizado, aun así, es lo habitual en el uso de plataformas No-code.

3.4.7 Validación para consultar candidatos

Esta etapa consistió en realizar la comprobación mediante el link de prueba enviado por WhatsApp y se constató que la información presentada con respecto a los candidatos fueran los mismo de acuerdo con la base de los datos. Además, la evaluación del diseño nuevamente estuvo a cargo del UX/UI Designer, teniendo en cuenta los siguientes puntos en comparación del diseño versus el resultado final, donde 1 es poco semejante, y 5 cumple con el diseño.

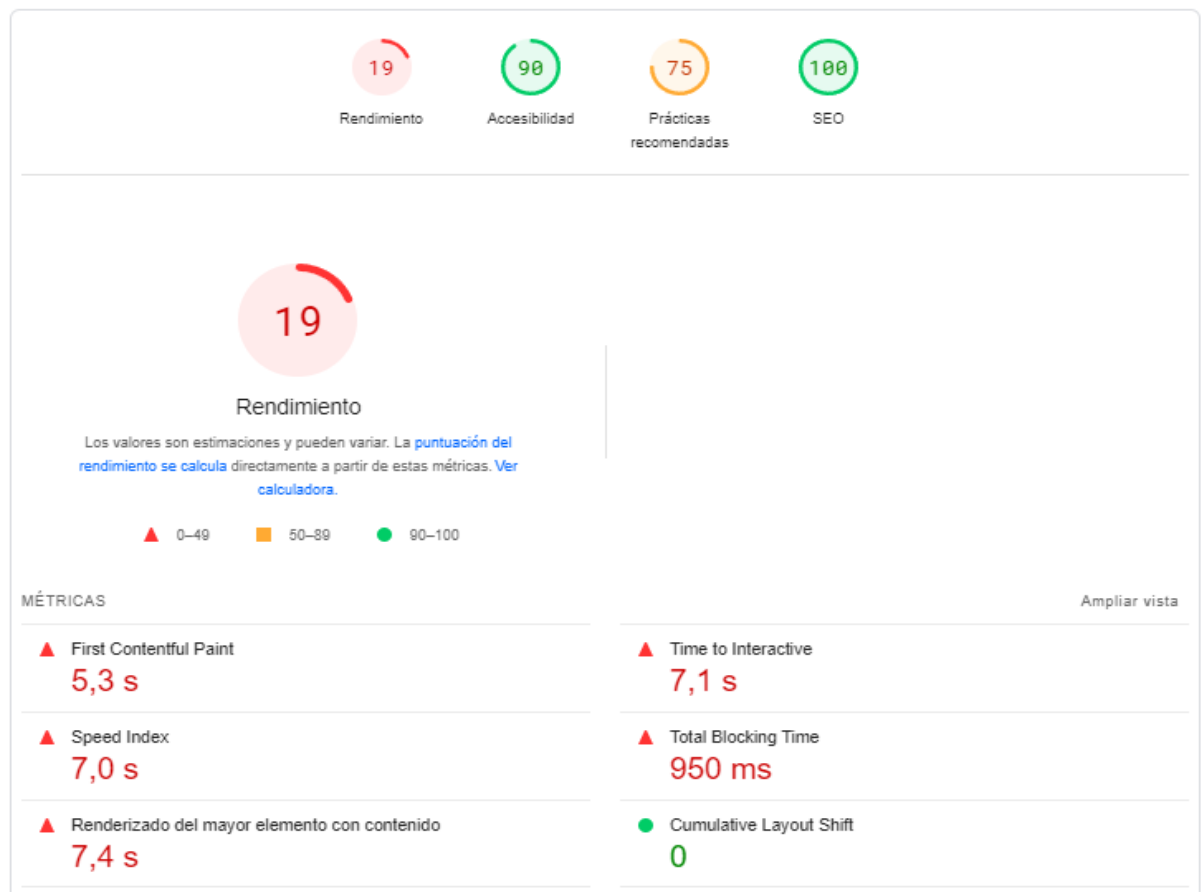
Tabla 14. Calificación de diseño consultar candidatos

Características	Similitud (1-5)
Colores	5
Proporciones y dimensiones	5
Fuentes y estilos	5
Responsividad para móvil	5

Fuente: Elaboración propia.

En la segunda etapa de validación de cara al usuario final, en este caso el reclutador, manifestó lo siguiente: “La información mostrada de los candidatos es clara, es fácil tener detalle de lo que posee cada perfil, la información es la necesaria para valorar y saber que posee, así como descargar su hoja de vida”.

Figura 35. Prueba de rendimiento consultar candidatos



Fuente: Elaboración propia. Herramienta de Google PageSpeed Insights

Se determino que, aunque la funcionalidad es accesible, puede tornarse un poco lenta su usabilidad en equipos de gama baja, aun así, es lo normal en el uso de plataformas No-code.

3.4.8 Validación para consultar vacantes

En la primera etapa de validación del caso de uso, en la subetapa funcional se abrió un enlace de prueba que fue enviado por WhatsApp y se comprobó con la base de datos que las vacantes que estaban siendo visualizadas, fuesen los mismas.

En la subetapa de diseño, este punto fue evaluado por un UX/UI Designer, quien es la persona que creó las distintas funcionalidades previamente a ser construidas y quien para la validación de diseño tuvo en cuenta los siguientes puntos en comparación del diseño versus el resultado final, donde 1 es poco semejante, y 5 cumple con el diseño.

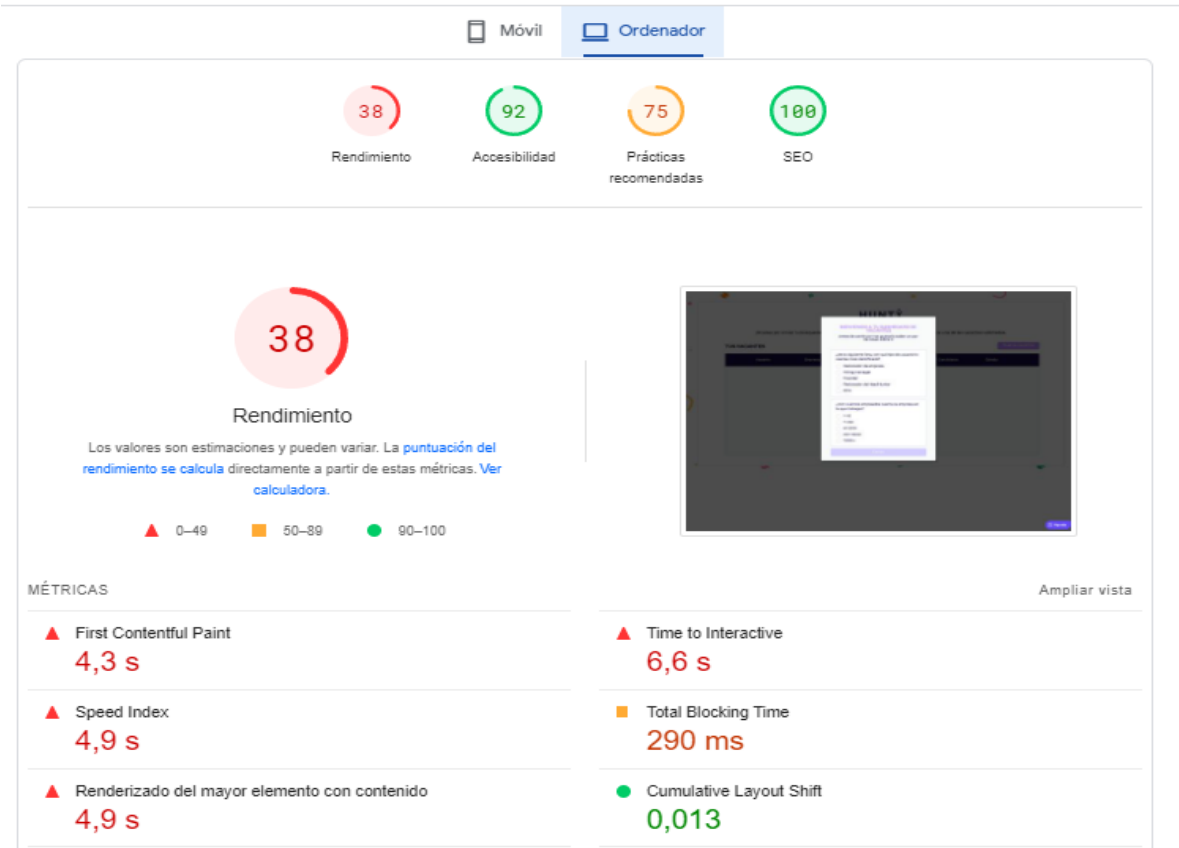
Tabla 15. Calificación de diseño consultar vacantes

Características	Similitud (1-5)
Colores	5
Proporciones y dimensiones	4
Fuentes y estilos	4
Responsividad para móvil	5

Fuente: Elaboración propia.

En el resultado el usuario final, dio a conocer que la información listada de las distintas vacantes registradas, fue puntual y precisa, presentaba claramente el estado de la vacante y la disponibilidad de los candidatos, o la interacción resumida con respecto a los mismos.

Figura 36. Prueba de rendimiento consultar vacantes



Fuente: Elaboración propia. Herramienta de Google PageSpeed Insights

Según los resultados, aunque la funcionalidad es accesible, puede ser lenta ya que esto depende de los equipos en los cuales se ejecute generalmente sucede en los de gama baja, sin embargo, es lo normal en el uso de plataformas No-code.

3.4.9 Validación para cambiar de estado

En este caso se ingresó a una vacante con recomendaciones enviadas y se usaron todos los botones que realizan cambios de estado en la recomendación, ya sea porque se agregó a favoritos un candidato, lo descartó o lo contactó; De igual forma se comprobó que en la base de datos se realizaron los cambios de manera exitosa.

Según el UX/UI Designer, quien fue el encargado de la evaluación pertinente acerca del diseño se pudo concluir los resultados teniendo en cuenta los siguientes puntos en comparación del diseño versus el resultado final, donde 1 es poco semejante, y 5 cumple con el diseño.

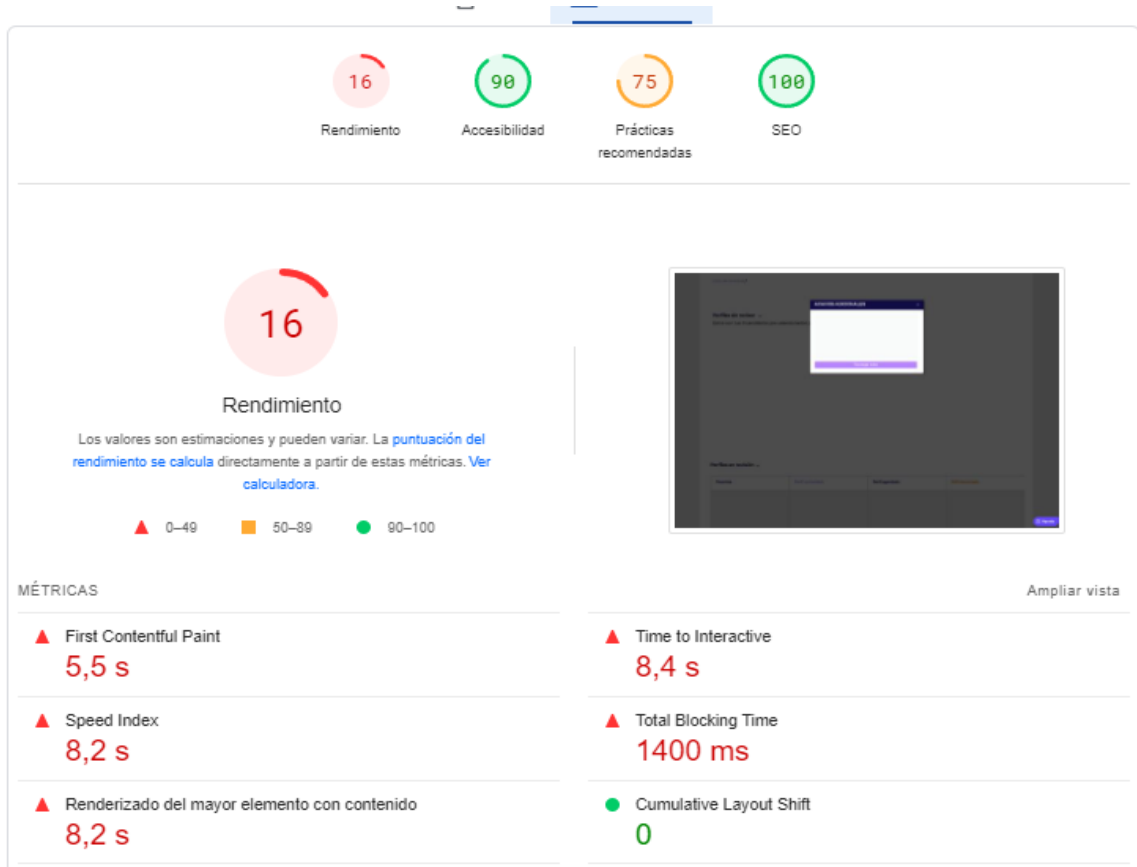
Tabla 16. Calificación de diseño cambiar de estado

Características	Similitud (1-5)
Colores	4
Proporciones y dimensiones	4
Fuentes y estilos	5
Responsividad para móvil	4

Fuente: Elaboración propia.

De acuerdo al usuario final la importancia de esta validación radicó en que se puede conocer las diferentes interacciones que pueda presentar cada candidato, así se tiene claridad de los candidatos que ya contactados o descartados.

Figura 37. Prueba de rendimiento cambiar de estado



Fuente: Elaboración propia. Herramienta de Google PageSpeed Insights

Se determinó que la funcionalidad es integral, pero puede tornarse un lenta dependiendo del equipo en el cual se usa, aun así es lo regular en el uso de plataformas No-code.

4 CONCLUSIONES

Durante la construcción del marco teórico se pudo notar que realmente de forma previa no existe una base teórica cien por ciento definida cuando se trata de establecer definiciones sobre algunos conceptos, esto debido a que el No-code es una nueva tendencia, una forma más reciente e innovadora de desarrollar un proyecto de Software. Realmente los antecedentes y proyectos previos no nos arrojan información exacta en un proyecto semejante. No obstante, basado en la experiencia personal se definió cada concepto con claridad.

Así mismo durante la descripción del desarrollo se pudo notar que realmente, aunque no hubo una ilustración concreta del código debido a las restricciones legales de la empresa donde se realizaron las prácticas, se percibió que evidentemente es mucho más ágil construir un producto software con una metodología ágil de desarrollo y una herramienta No-code.

Otro punto a concluir es que una persona incluso sin conocimientos previos sobre programación o desarrollo web podría ejecutar el proyecto, ya que su rápido e intuitivo aprendizaje hace que una empresa crezca sin mucho personal experto en desarrollo, de esta forma durante el desarrollo el proyecto, incluso uno de los fundadores de la empresa llevo a cabo otra funcionalidad sin esta ser su área en la que se desempeña. Si bien Bubble, la herramienta escogida, es un poco costosa, no lo sería tanto como si se hubiese contratado un equipo completo de desarrollo, además que con el transcurrir del tiempo, hubiese conllevado a extender todo hacía los tiempos tradicionales que se conocen, lo cual no hubiese sido una opción apropiada para esta Startup que se encontraba en una etapa crucial en el ámbito económico.

Dentro de la validación del desarrollo, se nota que ya puesto en funcionamiento la etapa de pruebas y lanzamiento final, el desarrollo web No-Code orientado a la administración de vacantes para una agencia de empleo se comporta de forma eficiente registros no superiores a 200, cuando esta cifra comienza a incrementarse, algunas funcionalidades se comportan de forma lenta, por lo que se concluye que la mayor desventaja de Bubble como herramienta No-Code es la optimización, no obstante, existen extensiones o plugin que optimizan la obtención de búsquedas, sin embargo resultaría un poco costoso hacerlo. Por lo que a medida que crezca el número de usuarios de la empresa, se estima que se realizará una migración a código tradicional.

5 BIBLIOGRAFÍA

CARACTERASTICAS DE un sistema EHR para hospitales | Cirrus [Anónimo]. Cirrus de Ecaresoft | Software moderno para Hospitales [página web]. [Consultado el 15, diciembre, 2022]. Disponible en Internet: <<https://www.getcirrus.com/page/caracteristicas-de-un-sistema-ehr-para-hospitales>>.

BACKEND Y Frontend, ¿Qué es y cómo funcionan en la programación? [Anónimo]. Conectividad y Soluciones de TI | Servnet [página web]. [Consultado el 18, diciembre, 2022]. Disponible en Internet: <<https://www.servnet.mx/blog/backend-y-frontend-partes-fundamentales-de-la-programacion-de-una-aplicacion-web>>

CLICKUP - Snoop Consulting - Partners [Anónimo]. Snoop Consulting [página web]. [Consultado el 18, diciembre, 2022]. Disponible en Internet: <<https://www.snoopconsulting.com/partners/clickup/#:~:text=ClickUp%20es%20una%20solución%20intuitiva,de%20desarrollo,%20marketing%20o%20ventas.>>.

COLABORADORES DE LOS PROYECTOS WIKIMEDIA. Google Analytics - Wikipedia, la enciclopedia libre. Wikipedia, la enciclopedia libre [página web]. (12, julio, 2007). [Consultado el 15, diciembre, 2022]. Disponible en Internet: <https://es.wikipedia.org/wiki/Google_Analytics>.

¿CÓMO SELECCIONAR un marco de desarrollo de software efectivo? | AppMaster [Anónimo]. AppMaster - The no-code platform for building web & mobile apps [página web]. [Consultado el 15, diciembre, 2022]. Disponible en Internet: <<https://appmaster.io/es/blog/marco-efectivo-de-desarrollo-de-software>>.

EL KEY account manager y su papel en las ventas de la empresa | Conexión ESAN [Anónimo]. ESAN Graduate School of Business - ESAN [página web]. [Consultado el 15, diciembre, 2022]. Disponible en Internet: <<https://www.esan.edu.pe/conexion-esan/el-key-account-manager-y-su-papel-en-las-ventas-de-la-empresa>>.

FERNANDEZ, Cristina & Benavides, Juan. 2020. Las plataformas digitales, la productividad y el empleo en Colombia. Bogota : s.n., 2020. pág. 66.

HALLBERG, Agnes. 2021. Using Low-Code Platforms to Collect Patient-Generated Health A Software Developer's Perspective. junio de 2021. págs. 1 - 13.

INTRODUCTION - Bubble Docs. Introduction - Bubble Docs. [En línea] [Citado el: 23 de Abril de 2022.] <https://manual.bubble.io/new-start-here/the-application-editor>.

IONOS Digitalguide. [En línea] [Citado el: 28 de febrero de 2022.] <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/que-es-kanban/>.

MANUAL DE Buble, Introduction - Bubble Docs. [En línea] [consultado el: 23 de Abril de 2022.] <https://manual.bubble.io/new-start-here/the-application-editor>.

MIXPANEL [Anónimo]. Marketing Market [página web]. [Consultado el 15, diciembre, 2022]. Disponible en Internet: <<http://marketingmarket.es/mixpanel>>.

PASTOR, Javier. Qué es Cloudflare, cómo funciona y por qué cuando "se cae" parte de internet se viene abajo. Xataka - Tecnología y gadgets, móviles, informática, electrónica [página web]. (2, julio, 2019). [Consultado el 18, diciembre, 2022]. Disponible en Internet: <<https://www.xataka.com/servicios/que-cloudflare-como-functiona-que-cuando-se-cae-parte-internet-se-viene-abajo>>.

PARRA AREVALO, Juan Pablo. 2020. Análisis del desarrollo de software en no desarrolladores. (diciembre,2020) # pag. 34 [Consultado: 23 de septiembre de 2021]. Disponible en: [http://caoba.sanmateo.edu.co/jspui/bitstream/123456789/231/3/\(1\)%20Analisis%20del%20desarrollo%20de%20software%20Proyecto_.pdf](http://caoba.sanmateo.edu.co/jspui/bitstream/123456789/231/3/(1)%20Analisis%20del%20desarrollo%20de%20software%20Proyecto_.pdf).

PEREZ, Lucia. 2021. La herramienta para crear apps sin tocar código. [En línea] 2021 de junio de 2021. [Citado el: 28 de marzo de 2022.] <https://bloo.media/blog/bubble-io/>.

PORTAFOLIO. 2017. Portafolio.co. [En línea] 06 de Septiembre de 2017. <https://www.portafolio.co/economia/empleo/por-que-para-muchos-colombianos-es-dificil-conseguir-trabajo-509432>.

¿QUÉ ES API RESTful? | Guía sobre API RESTful para principiantes | AWS [Anónimo]. Amazon Web Services, Inc. [página web]. [Consultado el 18, diciembre, 2022]. Disponible en Internet: <<https://aws.amazon.com/es/what-is/restful-api/>>.

QUÉ ES el Backend de una web y por qué es tan importante [Anónimo]. Rafa Arjonilla [página web]. [Consultado el 18, diciembre, 2022]. Disponible en Internet: <<https://rafarjonilla.com/que-es/backend/>>.

¿QUÉ ES Figma y para qué sirve? [Anónimo]. CEI: Escuela de Diseño y Marketing [página web]. [Consultado el 15, diciembre, 2022]. Disponible en Internet: <<https://cei.es/que-es-figma/>>.

¿QUÉ ES un plugin y para qué sirve? - El blog de dinahosting [Anónimo]. El blog de dinahosting [página web]. [Consultado el 18, diciembre, 2022]. Disponible en Internet: <<https://dinahosting.com/blog/que-es-un-plugin-y-para-que-sirve/>>.

SARACLIP. 2017. Requerimientos de un proyecto. [En línea] 15 de junio de 2017. [Citado el: 2022 de marzo de 2022.] <https://www.saraclip.com/requerimientos-de-un-proyecto/>.

SEO: La guía completa para que alcances la cima de Google [Anónimo]. Rock Content - ES [página web]. [Consultado el 15, diciembre, 2022]. Disponible en Internet: <<https://rockcontent.com/es/blog/que-es-seo/>>.

SLUG. Definición y cómo optimizarlo para SEO - Pululart [Anónimo]. Pululart [página web]. [Consultado el 15, diciembre, 2022]. Disponible en Internet: <<https://pululart.es/blog/slug-optimizarlo-seo/#:~:text=El%20slug%20es%20una%20herramienta,una%20o%20más%20palabras%20clave.>>>.

STARTUPEABLE. STARTUPEABLE. [En línea] <https://startupeable.com/glosario/no-code/>.

WOO, Marcus. 2020. The Rise of No/Low Code Software Development—No Experience Needed? [En línea] septiembre de 2020. [Citado el: 15 de febrero de 2022.] <https://www.sciencedirect.com/science/article/pii/S2095809920301843?via%3Dihub>.

ZAHUANTITLA Salas, Maribel. ETAPAS DEL desarrollo software - Fundamentos de Ingeniería de Software. [En línea] [Citado el: 28 de marzo de 2022.] <https://sites.google.com/site/fundamentosdeingendsoftware/u1-fundamentos-ingenieria-de-software/1-3-etapas-del-desarrollo-software>.